THE UNIVERSITY OF ALBERTA

A COMPUTERIZED TYPING SYSTEM FOR THE HANDICAPPED

by

© TERRY W. ENGELHART

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF MASTER OF SCIENCE

DEPARTMENT OF ELECTRICAL ENGINEERING

EDMONTON, ALBERTA

FALL, 1971.

THE UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled A Computerized Typing System for the Handicapped, submitted by Terry W. Engelhart in partial fulfilment of the requirements for the degree of Master of Science.

Date....June 11th 1971

ABSTRACT

This thesis describes the design and construction of a multi terminal typing system for the handicapped controlled by a small general purpose computer. It was designed to provide disabled persons, in particular paraplegics and quadraplegics, with a method of typing requiring only activation of a single switch.

Each patient was provided with a switch mechanism tailored to his disability such that a contact closure could be initiated reliably. The switch was used by him to control an automatic scanning process of the characters and words available for typing. To facilitate this a display unit was incorporated which individually displayed the characters and words by means of a photographic transparency.

The computer was used to originate all system control action. It was programmed to detect and act on patient commands, control scanning action on the display units, initiate typewriter operation. Each patient terminal was controlled seperately by the computer, multiplexing this control on a time sharing basis.

This method provides a large number of patients with access to a typing system at a relatively low cost and with greater flexibility than possible with single typewriter systems using a hardware approach.

ACKNOWLEDGEMENTS

The author wishes to take this opportunity to thank his supervisor, Professor Y.J. Kingma, for his generous advice, assistance and encouragement.

He also wishes to thank other members of the staff for their suggestions and assistance, particularly, P.J.R. Harding.

The author also expresses his appreciation to his wife for her patience and understanding and to the Department of Engineering for financial assistance received throughout the term of this thesis.

TABLE OF CONTENTS

Page

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

In recent years a number of devices and systems have been developed to assist the severely handicapped to communicate and perform small tasks. One of the more successful approaches has been typewriter orientated systems which enable the patient to type by means of a suitable electronic interface.

This project demonstrates the possibility of using a small general purpose computer as the main control element in a multi-station typing system for paraplegic and quadraplegic patients. Use was made of the speed and flexibility of a computer, which enabled a number of patients to type simultaneously on seperate typewriters, and also provided a more sophisticated mode of operation over techniques using seperate control hardware for each patient.

The design of this system provided each patient with a terminal consisting of a display unit, input device and output writer. From the point of view of patient operation this approach is similar to that taken by Mr. Kildaw[6] for single typewriter systems. Each patient terminal was designed to be connected to a remote control centre consisting of a Digital Equipment Corporation PDP-8 computer and a central interface unit. An effort was made to keep the amount of hardware to a minimum and make use of the computer's capabilities as much as possible.

The display unit enables the patient to see the characters, words, controls and special functions available to him. The display consists of a back lit film transparency which, in conjunction with appropriate computer program changes, can be altered to provide various control formats for the patient. For the purpose of this project only one transparency was provided to demonstrate the flexibility and versatility of this approach. The transparency displays all the characters and controls of a typewriter, plus a group of words which can be typed in one control sequence to demonstrate the possibility of producing a faster and more sophisticated means of patient communication. It also displays a control output for a peripheral device to show the feasability of controlling a number of such devices from the patient's terminal.

The input device consists of a simple switch adapted to the disability of the patient such that it can be activated reliably to provide a control input to the system. The control input allows the patient to select the character or word to be typed or function performed. This device is the only aspect of the system which must be tailored to a particular patient or type of disability. It was not the purpose of this project to develop a series of input devices, therefore it was only attempted to indicate the various possibilities and define the system requirements in this regard.

The patient output writers are IBM Selectric typewriters equipped with special electromechanical actuators, which reduced

the cost relative to the IO writers normally used for these systems. The actuator was designed to be easily removable, resting on the keyboard and necessitating no modifications to the typewriter which can therefore be used for other purposes.

A central interface unit was designed to provide all the necessary logic to facilitate transfer of patient control data into the computer and computer output commands to the patient terminals.

The system program developed for this project provides for two terminal operation with expandability to 42. The program was structured to enable further terminals to be added without extensive program alterations. Machine language programming was used to optimize the terminal capacity for the core memory size of the computer.

It is hoped that this project will provide the design parameters to enable establishment of large typewriter systems for use in hospitals. These systems provide patients with a means of communication and the incentive to gain better control of their remaining faculties.

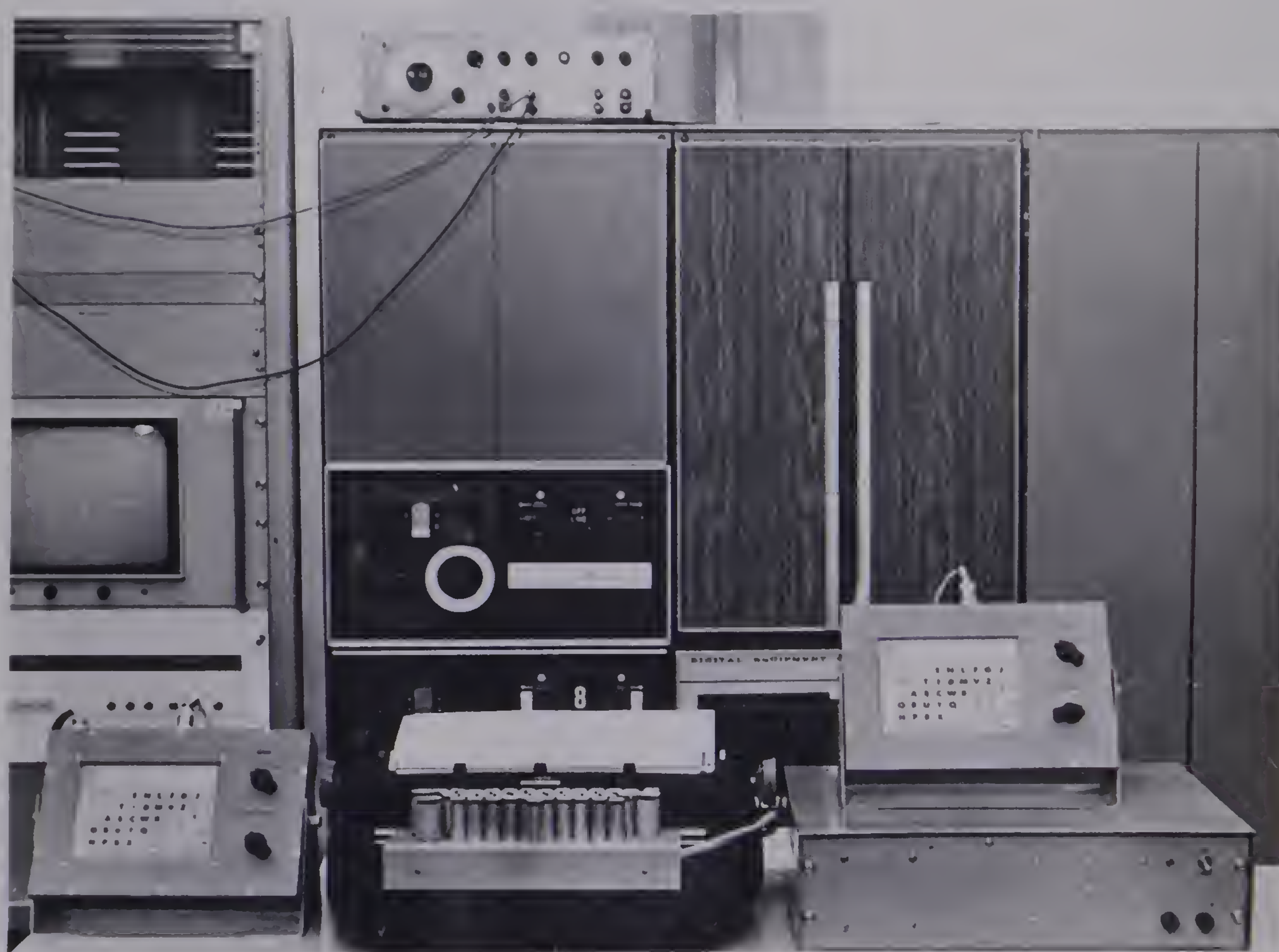
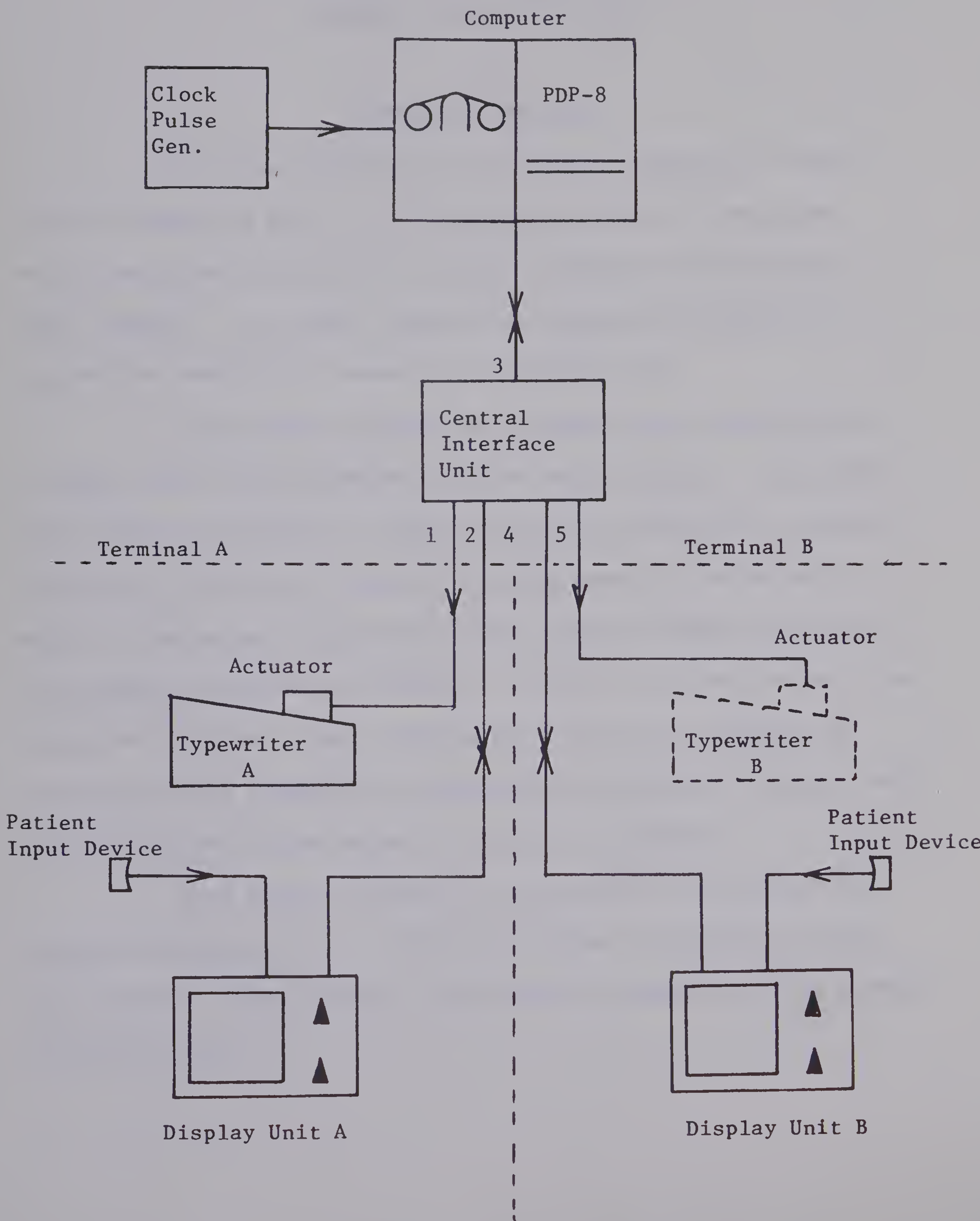


Fig. 1.1 The Complete Typing System

Fig. 1.2 Total System Flow



CHAPTER 1

GENERAL SYSTEM DESCRIPTION

and

OPERATING PROCEDURE

The block diagram for the complete typewriter control system is shown in Fig. 1.1. The system provides for two typewriter operation controlled by a Digital Equipment Corporation PDP-8 computer. At present typewriter operation on terminal 'B' can only be simulated by means of the display unit.

Each patient terminal is equipped with a display unit, an input device and a solenoid actuated output writer. The display unit enables the patient to select the mode of operation, scan rate and letter or word to be typed. The input device is selected to adapt to the patient's disability and provides a digital input to the computer, enabling the patient to control the selection and final typing of a letter or word. The output writer is a standard IBM Selectric office typewriter, equipped with a removable solenoid overlay which allows either manual or automatic actuation.

Each patient terminal is connected to the computer via a central interface unit. This unit provides all necessary interface hardware, power supplies and typewriter solenoid driving circuits for both systems.

The computer initiates all necessary control signals associated with scanning and typewriter actuation. Only a fraction of the computer memory capacity of 4096 words is being used for this system. The existing program could be extended to handle an additional 42 terminals with the present modes of operation.

The display unit is a visual aid to allow the patient to monitor computer action and initiate appropriate control signals. It also acts as a control terminal to allow selection of mode and scan rate.

The display portion of the unit consists of an array of 62 "back lit" windows illuminating particular characters, words or control functions. With the exception of "shift" only one window can be illuminated at a time. By effecting the appropriate control signals the patient can access the desired window and type the character or word depicted.

The first column of the array, with the exception of "home" and "start", is blank, and acts as a buffer to allow sufficient patient reaction time between selection of a row and selection of the character in the adjacent column. The "home" position is a static, or rest, position to which the unit will return after a control sequence and remain until a new one is initiated. The "start" position is similar to the blank column, providing a buffer time between initiating a control sequence and selection of the first row.

The windows comprising rows 3 to 7 and columns 2 to 11 display all the characters and controls of an IBM Selectric typewriter. The only exception is "SW" which provides a switch closure to control a peripheral device. The characters are not arranged in the format of the typewriter or in alphabetical order, but in such a way as to provide minimum access time for characters used most often.[4,5]

The "shift" position is the only window that is illuminated simultaneously with another. Upon activation of "shift" the unit latches up, illuminating "shift" continuously and returning to the "home" position. The character called for in the next control sequence will be typed in upper case after which the "shift" lamp will be disabled and all subsequent characters will be in lower case.

The bottom row of windows display frequently used words.[4,5] They are arranged in such a way as to provide minimum access time. When a word is activated the computer ignores any further control signals and proceeds to type the word followed by a space. As each letter of the word is being typed the appropriate window is illuminated. If "shift" was selected prior to the word the first letter of the word would be in upper case while the remaining would be in lower case. There is no provision for upper case typing of the entire word.

The patient controls the system by activating the input device which provides a momentary or continuous contact closure.

This signal is fed into a connector at the top of the display unit. The computer acts upon this signal in different ways depending on the setting of the "mode" selector switch.

The "mode" switch has four settings, "S", "D", "CS" and "CD" which determine the method by which the patient can select a character.

The setting "S" provides a single acting scan mode. The computer interprets a change of switch state from open to closed as a command. The change of state from closed to open is ignored. To select and type a character four command signals must be activated. The first command initiates automatic scanning of the first column to select the desired row. When the row is reached a second command is required to scan the row for the desired character. When the character is reached a third command is needed to stop the scanning process and await the final command which types the illuminated character and resets the display to "home". In the event of a selection error being made the patient can reset the system to "home" in three ways. If the desired row is missed the system will reset at the end of column one providing no further commands are initiated. Should the incorrect row be chosen or the desired character missed, the system will reset at the end of the row providing no further commands are given. If the incorrect character is selected the system will reset to "home" after three scan intervals if no type

command is initiated.

The "S" mode is quite tolerant of erratic patient operation. After each command the program waits one full "scan rate" time interval before proceeding to the next state. Therefore double commands and selection errors can be avoided by setting the scan rate to a speed compatible with the patients capabilities.

The "D" setting provides a double acting mode. The computer interprets any change of input state as a command. Therefore both switch closures and switch openings initiate control commands. The same four command signals are required to select and type a character as with the single acting mode. In order to produce the four commands the patient need only activate the input device twice, but must maintain activation between commands.

The same methods as described for the single acting mode can be used in this mode to reset the system in the event a selection error is made.

This mode greatly increases typing speed over the single acting mode but sacrifices system tolerance to erratic patient operation. Although the program waits a full scan interval after each command before proceeding to the next state, selection errors will occur if the patient can not maintain activation of the input device between commands.

The "CS" mode setting provides a continuous scan with a single acting input mode. As with the previously discussed "S" mode,

the computer interprets a change of switch state from open to closed as a command. In order to select and type a character only three commands are required. The first initiates scanning, which proceeds continuously from left to right, starting with the top row through to the last word of the bottom row. When the desired character is reached a second command is required to stop the scanning process to await a final command which types the character and resets the display to "home". If a selection error is made the patient can reset the system to "home" in two ways. Should the desired character be missed the system must scan to the end of the last row to reset. If the incorrect character is selected the system will reset to "home" after three scan intervals if no type command is given.

Mode setting "CD" provides continuous scanning with a double acting mode. This combines the double acting input detection used in the "D" mode with continuous scanning selection techniques described in the "CS" mode. As before three commands are required to select and type a character. Scanning commences with the activation of the input device and must be maintained until the desired character is reached. At this point it is released and activated again to type the character. This control sequence can be reset to "home" in the same manner described for the "CS" mode, in the event a selection error is made.

The "scan rate" selector switch of the display unit allows

the patient to put his terminal in service and select the scan rate to suit his capabilities. The "off" position puts the terminal in an off line state, directing the computer program to skip over the program controlling the terminal. The other selector positions 1 through 6 select the scan rate. Each number defines the number of unit time intervals required for the system to progress from one window to the next. A time interval is defined by the clock pulses associated with the computer. This has been programmed such that 1000_8 clock pulses represent one unit time interval. The signal generator producing the clock pulses was set at 1 K Hz compatible with program cycle times. Therefore at a clock rate of 1 millisecond a unit time interval is 0.512 seconds. The maximum scan rate that can be programmed is only limited by the computer program cycle time which is 2 milli seconds for two patient terminals and 42 milli seconds for the maximum of 42 terminals. This of course far exceeds the reaction speed of patients. Words are programmed to type letter for letter at a fixed (programmed) rate of 5 characters per second, but could be increased to a maximum of 8 characters per second if desired.

The central interface unit contains two power supplies, typewriter solenoid driving circuits and all computer interface hardware. All interconnections between the computer and the two patient terminals are made through this unit. Provision for six cable connectors has been made at the rear of the unit and are

designated 1 through 6 starting at the left. Connectors 1 and 4 connect the display units of patient terminals "A" and "B" respectively, consisting of lamp control lines, lamp supply voltage and switch control lines. Connectors 2 and 5 connect the typewriter actuators for patient terminals "A" and "B" respectively. Connector 3 interconnects the system with the computer, with device selector strobe lines and accumulator input and output buses. Connector 6 is spare and not used for this project.

Two seperate power supplies are provided supplying three voltages for lamps, solenoides and logic circuitry. The supplies are switched on and off by one main switch on the front of the interface unit. Beneath the switch are two fuses marked F1 and F2 for power supplies 1 and 2 respectively. Power supply 1 is unregulated and supplies voltages of 56 VDC and 26 VDC for typewriter solenoids and display lamps respectively. Power supply 2 is fully regulated and provides 5 VDC for all interface logic. The fuses installed in F1 and F2 are rated at 1 amp and 1/8 amp respectively and are 3 AG "slow blow" types. The primary voltage for the supplies must be connected to that of the computer, so that when the computer is turned off the power supplies are also turned off. This is necessary to prevent erroneous computer codes causing continuous actuation of typewriter solenoids, which are only designed for intermittant duty.

The logic of the central interface unit consists primarily of Digital Equipment Corporation 'M' Series. The logic is used for decoders, buffer registers, driving circuits and timing circuits to completely interface and multiplex the two patient terminals to the computer. The present unit only has capacity for two terminals, if the system is to be extended a suitably larger unit would be necessary.

The typewriter actuator consists of 46 solenoids and a decoding network mounted in a frame designed to fit over any IBM Selectric typewriter. There is a solenoid associated with each character and control on the keyboard with the exception of "tab set" and power on/off. Therefore before placing the actuator on the typewriter the margins and tabulators must be set and the power turned on. The actuator is designed to press-fit over the keyboard and is held in place by virtue of it's own weight. Therefore the device can be removed with ease and the typewriter used elsewhere when the system is not in service.

The typewriter actuator decoding network reduces the number of control lines required to actuate the solenoids from 46 to 17 by incorporating an X-Y access system. The attached cable containing these lines plugs into the rear of the central interface unit where the drive signals originate. The typewriter keys are actuated by applying a 50 ms control pulse to the appropriate X and Y control lines or by

operating them manually by depressing the top of the desired solenoid plunger.

The patient input devices that the present system can accept must be in digital form. They can take the form of switch closures, closed is a logical "1" and open a logical "0". In addition voltage pulses can be used, 5 VDC being a logical "0" and 0 VDC a logical "1". These inputs are directly connected to the inputs of TTL logic gates. Therefore input voltages and polarities must be in keeping with the TTL logic.

The input device must be selected to take advantage of the remaining capabilities of the patient. It was not the object of this project to develop a full range of input devices but a few types were tried, some of which were developed by other projects[6,7]. A simple micro switch, for example, can be provided with various mechanical linkages to be actuated by foot, leg, hand or arm. A pressure transducer activated by the patient blowing through a tube connected to it was tried. It consisted of a pressure sensitive transistor energizing a reed relay when sufficient pressure is applied. A voice actuated input device consisting of a microphone and a voice actuated relay proved quite useful in that any noise could initiate an input. Various types of photoelectric devices can be used by merely devising a light and photoelectric pick-up system such that the light beam can be interrupted by any part of the patients body.

The input device used must be connected to the system by means of the receptacle located on top of the patients display unit.

More sophisticated input devices could be made acceptable to the system by additions to the computer program. In particular coded inputs such as Morse Code could be processed.

The PDP-8 computer used for this project has a core memory capacity of 4,096 words. Programming was done in machine language and stored on magnetic tape. The control program occupies 851 words of memory space, with each terminal requiring 90 words to store terminal state information necessary to facilitate multiplexing. Multiplexing of the control program between the two terminals is accomplished by means of a signal generator (clock pulse generator) which provides a real time reference for the computer. The computer time shares activity between two terminals every clock pulse. The clock pulse rate is set at 1k Hz with each pulse initiating a control cycle, operating on each terminal alternately.

To put the system into operation the computer program, stored as systems program S:CTSH, must be transferred from tape to core memory. After the patient terminals have been placed in an operating mode and the power turned on at the central interface unit the typing system will proceed as discussed with no further assistance required by an attendant.

CHAPTER 2

SYSTEM HARDWARE

The hardware for this system consists of a typewriter and typewriter actuator for each terminal, a display unit for each terminal and a central interface unit.

In keeping with the objectives of this project the hardware was kept to a minimum, providing only the necessary man to machine (typewriter and display unit) and hardware to software (central interface unit) interfacing.

Fig. 2.1 Typewriter - Typewriter Actuator

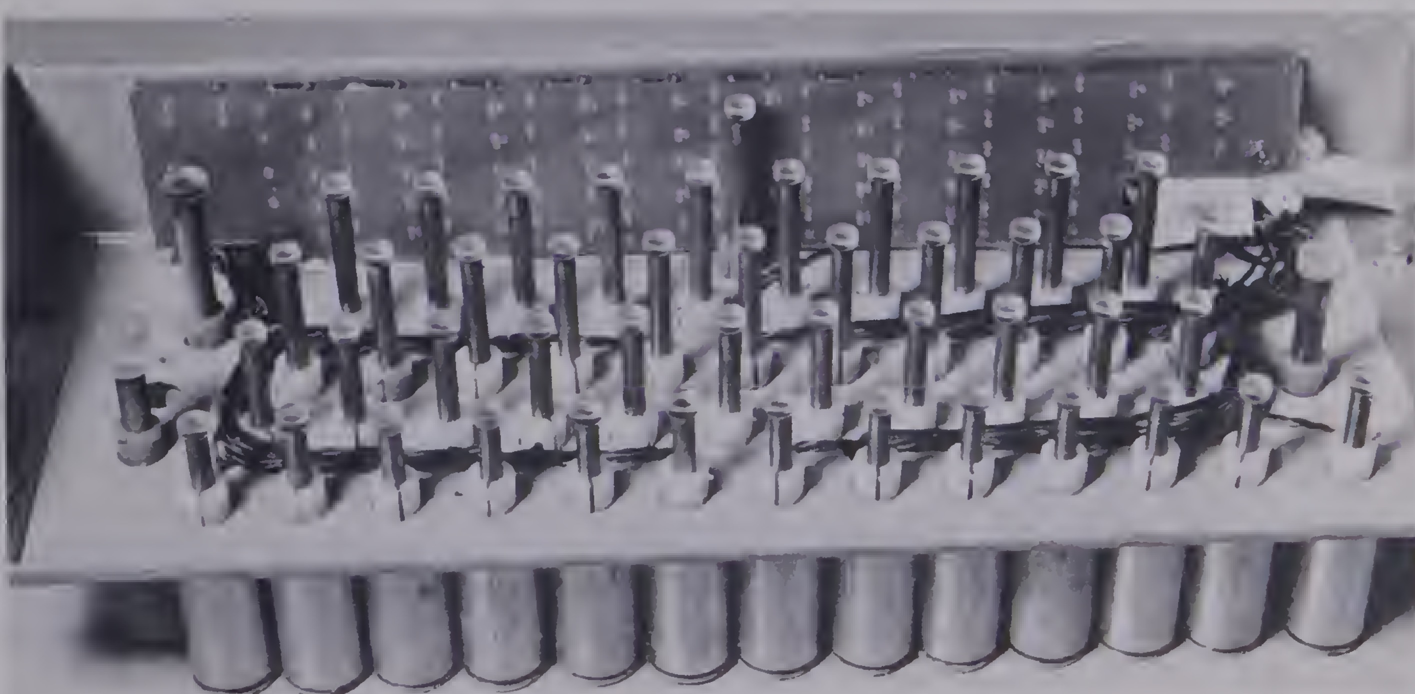


Fig. 2.2 Typewriter Actuator (Bottom View)

2.1 Typewriter Actuator

Each patient terminal requires an output writer. In order to accomplish this economically a solenoid overlay was designed which rests over the keys of a standard IBM Model 70 Selectric typewriter, operating the keys electromechanically.

Figs. 2.1 and 2.2 show photographs of the actuator which consists of a mounting plate, solenoid for each key and a diode decoding matrix.

2.1.1 Solenoid Mounting Plate

The mounting plate consists of a piece of 16 gauge steel, formed to conform to the contour of the typewriter's housing, with two alignment tabs to hold it in place. The plate has a 3/8 inch solenoid mounting hole corresponding to each key.

The alignment tabs are secured to either end of the mounting plate and are constructed so as to accurately position the actuator over the keyboard. The tabs protrude into the typewriter between the outside keys and the typewriter housing. In this way the actuator is held in an operating position by virtue of its own weight and the tabs.

This arrangement allows any Selectric typewriter to be converted into an output writer, with no structural modifications but merely by placing the actuator on the keyboard.

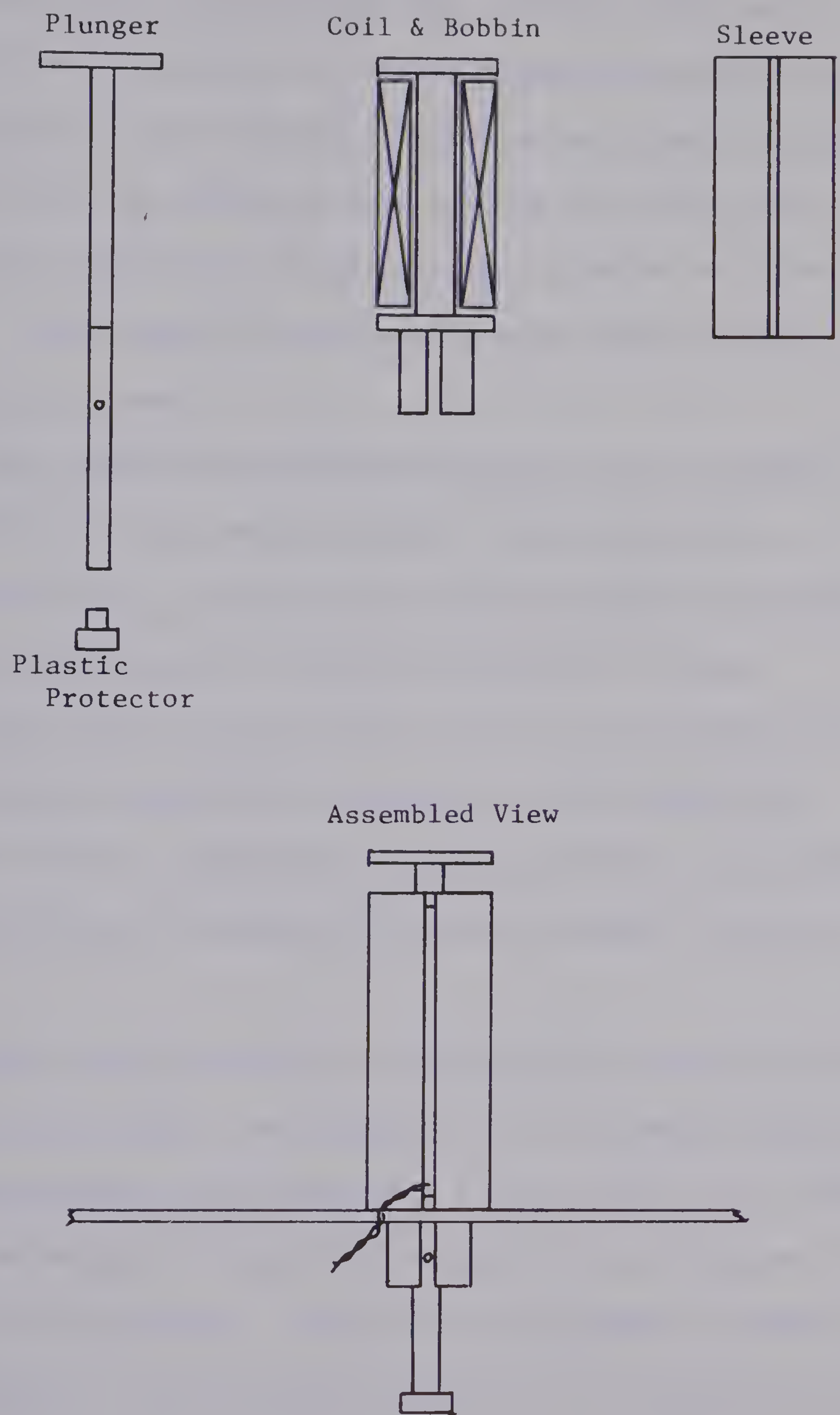


FIG. 2.3 Solenoid Assembly

2.1.2 Solenoids

No commercial solenoids were available which would satisfy the force requirements and allow convenient mounting on the close centres of the keyboard. Therefore two types of solenoids were designed, one for character keys and one for control keys.

The construction details of the solenoids are shown in Fig. 2.3. Each assembly consists of a nylon bobbin, coil, plunger and outer sleeve.

The bobbins are machined from nylon rod to provide a form for the coil and a mounting stud. The mounting stud portion of the bobbin is designed to protrude through the holes in the actuator mounting plate to hold the solenoids in place.

The stud is fitted with a slot in order to secure the plunger. This was accomplished by means of a pin through the plunger shaft which is positioned in such a way that it runs free in the slot during key actuation but prevents plunger rotation or removal.

The plunger consists of a steel disc, a steel rod and a length of brass tubing. The tubing is cut to a length appropriate to its key assignment and allowing for ¼ of an inch of key travel. The end of the tubing is fitted with a plastic plug acting as a bumper to protect the key surfaces. The top of each plunger is labled with its key assignment to allow manual operation of the typewriter with

the actuator in place.

The outer sleeve of the solenoid is a length of standard steel conduit. The sleeve completes the magnetic circuit of the coil and is slotted along its full length to eliminate an effective shorted turn through the sleeve. The sleeve was press-fit over the bobbin coil assembly and sealed with GE Silicon Seal. The coil wires are brought out through the slot in the sleeve and an auxilary hole in the mounting plate to the diode decoding matrix below.

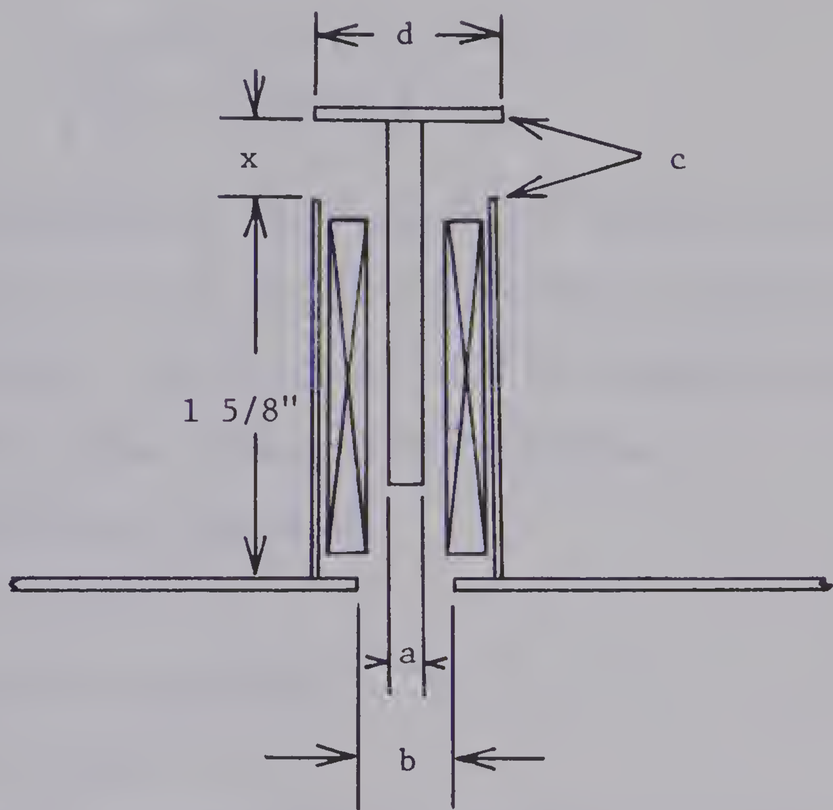
Key actuation force figures were not available for the Model 70 Selextric typewriter. Therefore static load tests were performed on the typewriter to obtain design figures for the solenoid coils. The results were as follows

Character Keys	:	2.7 ounces	(typical)
Control Keys	:	5.1 ounces	(typical)

These values are not adjustable and vary as much as 20% from key to key and for different typewriters.

It is realized that dynamic force characteristics would be more realistic. But since the required force varies and the application is not critical large margins were applied with little attention paid to optimizing the design.

The solenoid design can be analysed by a standard method for "singly excited electromechanical systems". A simplified diagram of the solenoid's magnetic circuit is shown in Fig. 2.4.



	x	a	b	c	d
Character Solenoids	1/4"	3/16"	3/8"	1/16"	3/4"
Control Solenoids	1/4"	1/4"	3/8"	1/16"	1 1/8"

FIG. 2.4 Simplified Solenoid Drawing

The force equation for this system is

$$f = -\tfrac{1}{2}\,\phi^2 \quad \frac{dR(x)}{dx}$$

For small displacements in "x" this can be reduced to the following

$$f = -\tfrac{1}{2} \qquad \frac{N^2 \quad I^2}{\left(\dfrac{x}{\mu_o \pi dc} + \dfrac{b - a}{\mu_o \pi c(a+b)}\right)^2 \mu_o \pi dc}$$

The dimensions of the solenoid are limited by the type-writer key spacing and the desire to use standard tubing and rod materials to simplify construction. With the dimensions shown in Fig. 2.4 the force formula reduces to the following

Character key solenoids

$$f = -N^2 I^2 \; 2.6\text{x}10^{-5} \text{ oz.}$$

Control key solenoids

$$f = -N^2 I^2 \; 3.2\text{x}10^{-5} \text{ oz.}$$

A number of solenoids were constructed and tested to arrive at the best compromise between size, number of turns and current. The final designs performed as shown in Table 2.1

TABLE 2.1

Minimum Continuous Solenoid Current

	Turns	I_{min}	R_{coil}	Force
	N	Amps	Ohms	Ounces
Character Solenoids	4000	0.09	150	3.4
Control Solenoids	3000	0.17	80	8.3

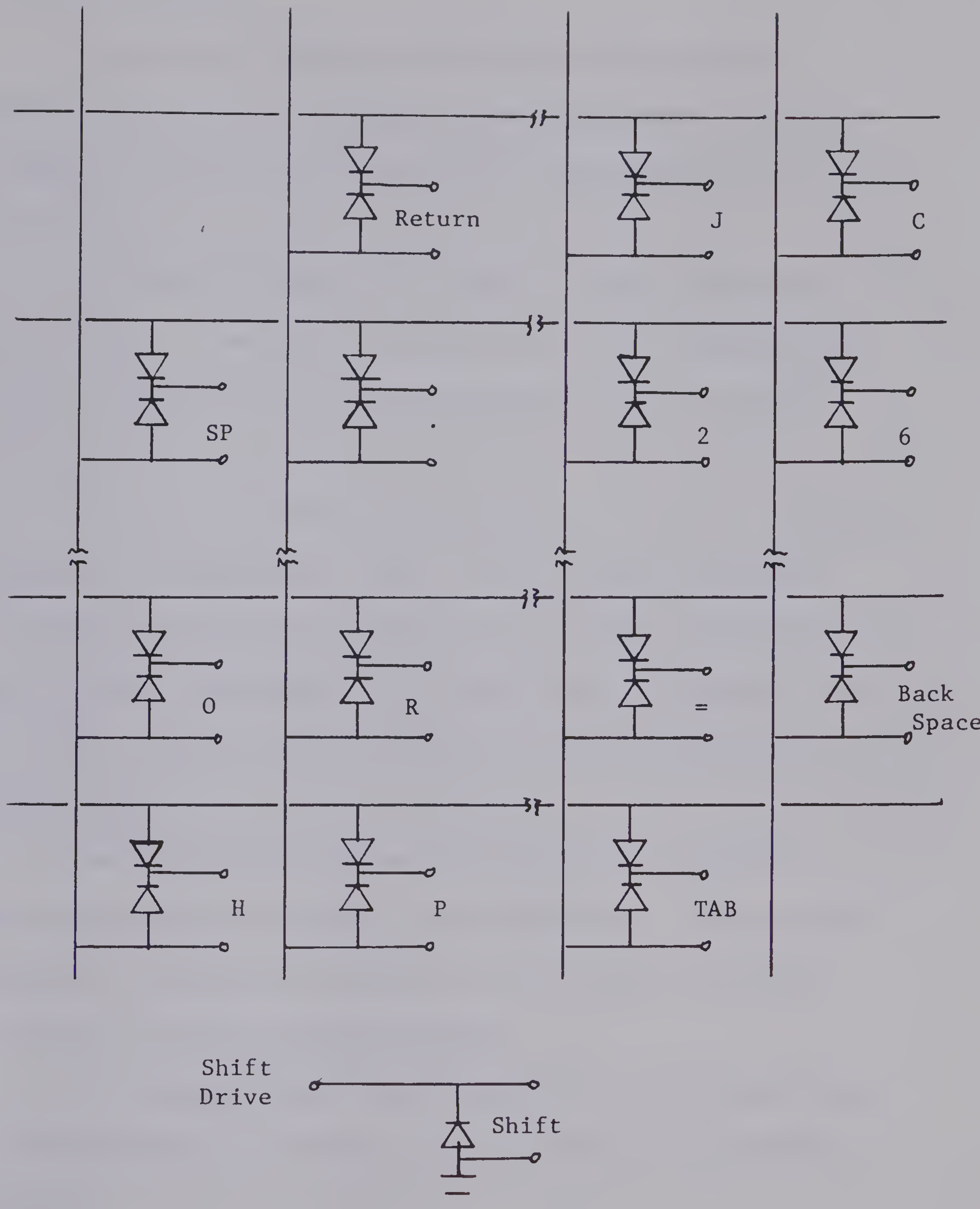
These actuating forces are somewhat greater than those predicted by the initial static tests on the keyboard. This is due to plunger friction and the relatively long stroke required.

A typewriter speed of 5 characters per second was selected as adequate for this application and within the capabilities of the actuator. Tests performed on the unit resulted in a choice of current pulse widths of 50 ms for character keys and 110 ms for control keys to provide reliable operation. Using these pulse widths it was found that the current had to be increased over that shown in Table 2.1 in order to ensure key actuation. The final drive pulses were as follows

TABLE 2.2

Solenoid Drive Pulses

	Pulse Width	I	V
	milli sec.	amps	volts
Character Solenoids	50	0.33	56
Control Solenoids	110	0.73	56



Note: All diodes are IN 4003

FIG. 2.5 Typewriter Activator Diode Matrix

2.1.3. Matrix Decoder

The circuit diagram for the diode matrix decoder is shown in Fig. 2.5. It was designed to reduce the number of control lines required to activate any one of the 44 characters and 5 control solenoids to 16.

The circuit requires two, IN4003, diodes associated with each solenoid. One diode is across the coil to suppress back emf voltages and the other is a decoding diode in an X-Y matrix decoding system.

Only one solenoid can be energized at a time by enabling the appropriate X and Y buses. The Y, or row, bus is energized with a +56 VDC pulse and its respective X, or column, bus is grounded, thereby activating the solenoid. The only exception to this is the "shift" solenoid which is separate from the matrix system and has its own control line.

The X-Y coordinates for each character or control solenoid corresponds to a window on the display unit. The two units can therefore be connected in parallel at the central interface unit to utilize the same encoding hardware.

The decoder circuit was constructed on a printed circuit board (PCB) mounted on the inside vertical surface of the solenoid mounting plate.

A control cable was hard wired to the PCB and was equipped with a cable connector to plug into receptacle CN-2 of the control interface unit.

Fig. 2.6 Display Unit

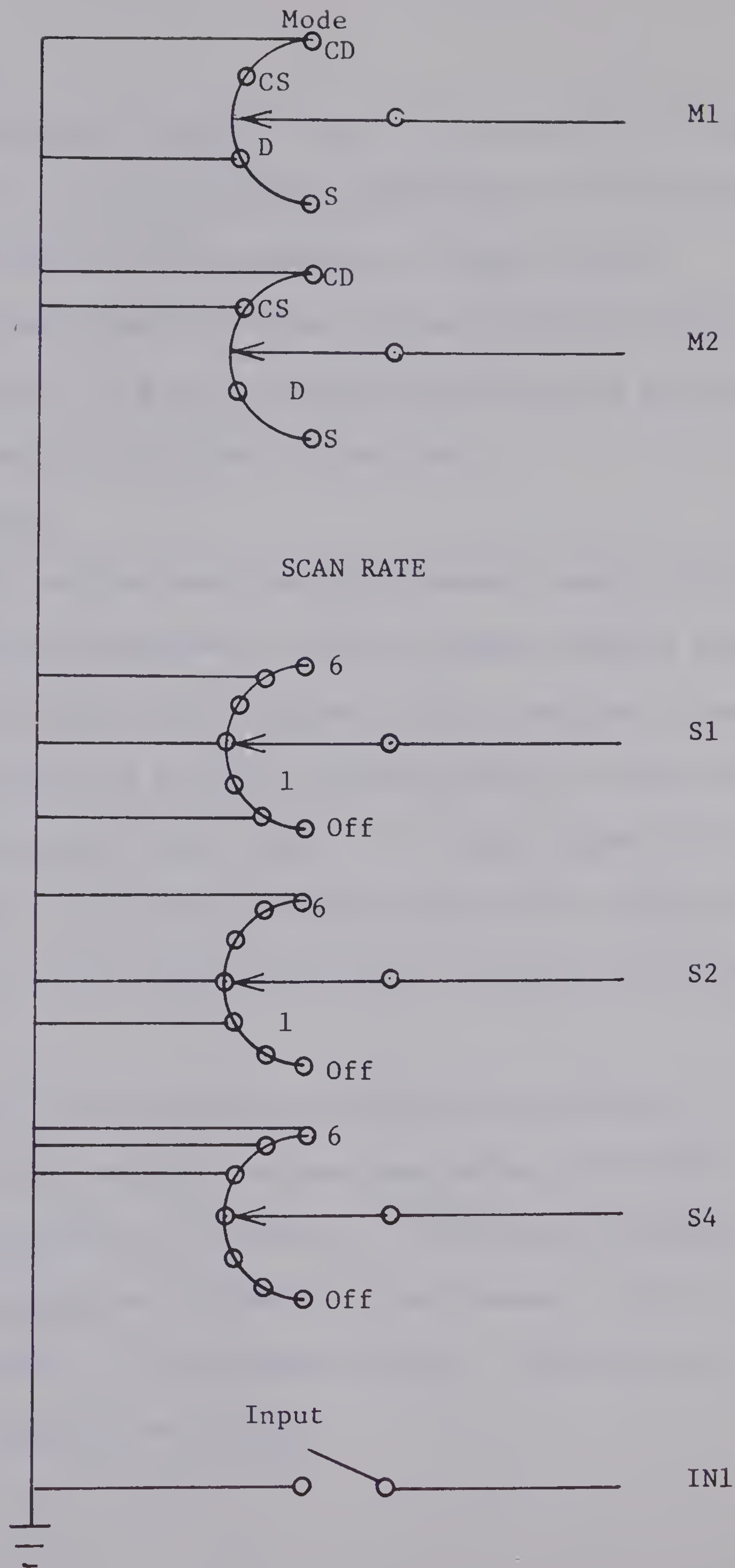


FIG. 2.7 Display Unit Output Circuits

2.2. Display Unit

The display unit shown in Fig. 2.6 provides the necessary visual and electrical interfaces between the patient and the control system. The unit consists of two sections, an input section (computer control signals) and an output section (patient display). The unit is housed in a 2 x 8 x 12 inch panel which can be mounted on a stand or clamped to the patient's wheelchair.

2.2.1. Input Section

The input section provides the necessary input to the computer to establish the terminal's operating mode, scanning action and initiate typing. The circuit diagram for this section is shown in Fig. 2.7 and consists of a "mode" encoding switch, a "scan rate" encoding switch and patient input device. The output from this circuit is in a binary code which is transmitted to the computer via the central interface unit using multiplexing techniques described in Chapter 3.

The codes from the patient terminals are stored as control variables in the software and are designated as AOI/ANI for terminal A and BOI/BNI for terminal B. The codes are established by energizing the appropriate accumulator input buses. A "0" is created by a ground and a "1" by an open circuit. The coding for both terminals is identical as follows.

TABLE 2.3

Display Unit Controls

Control Switch	Selection	Display Output Code					
		IN1	M1	M2	S1	S2	S4
Input	Off	0					
	On	1					
Mode	S		0	0			
	D		1	0			
	CS		0	1			
	CD		1	1			
Scan Rate	Off				0	0	0
	1				1	0	0
	2				0	1	0
	3				1	1	0
	4				0	0	1
	5				1	0	1
	6				0	1	0

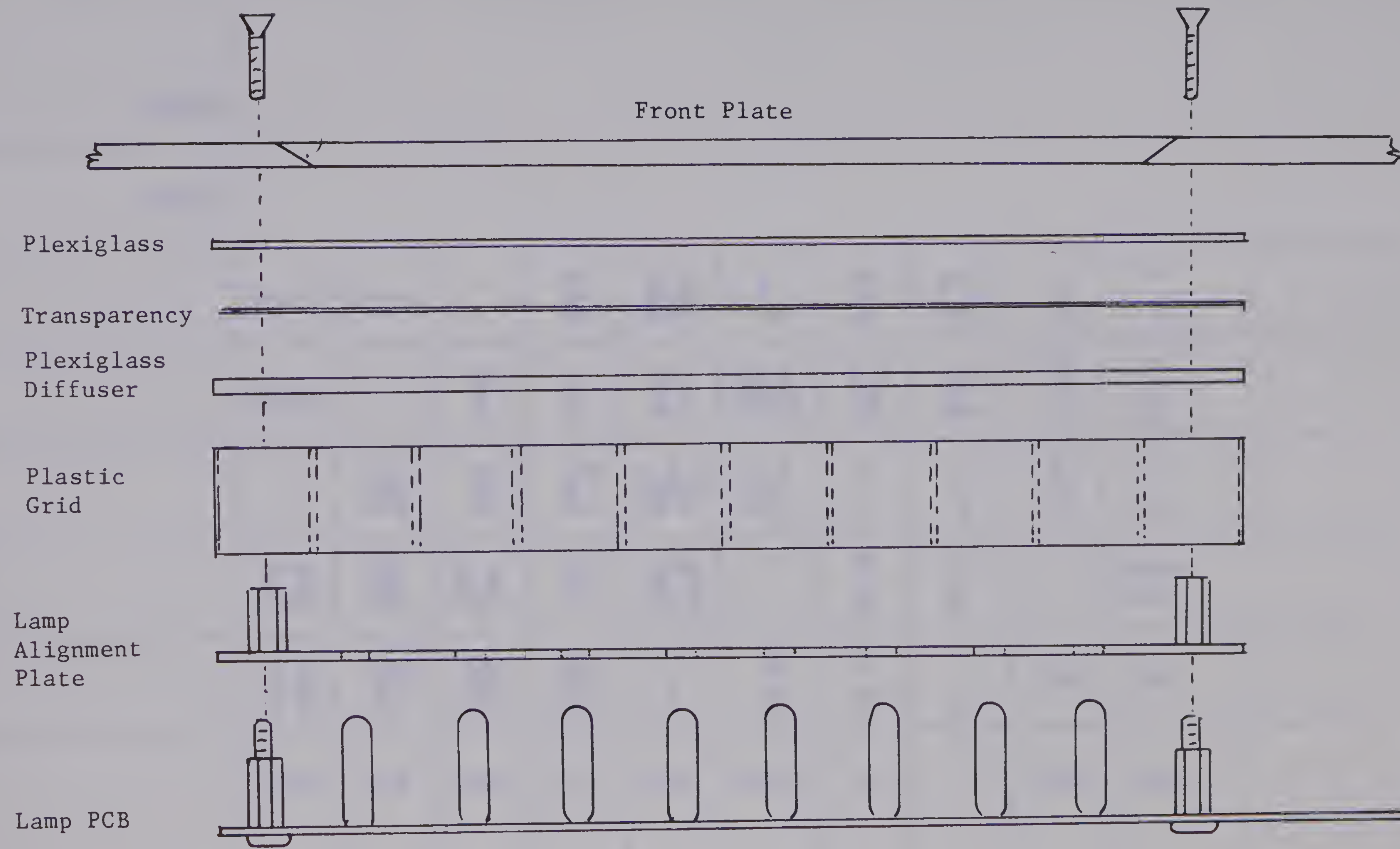


FIG. 2.8 Display Assembly

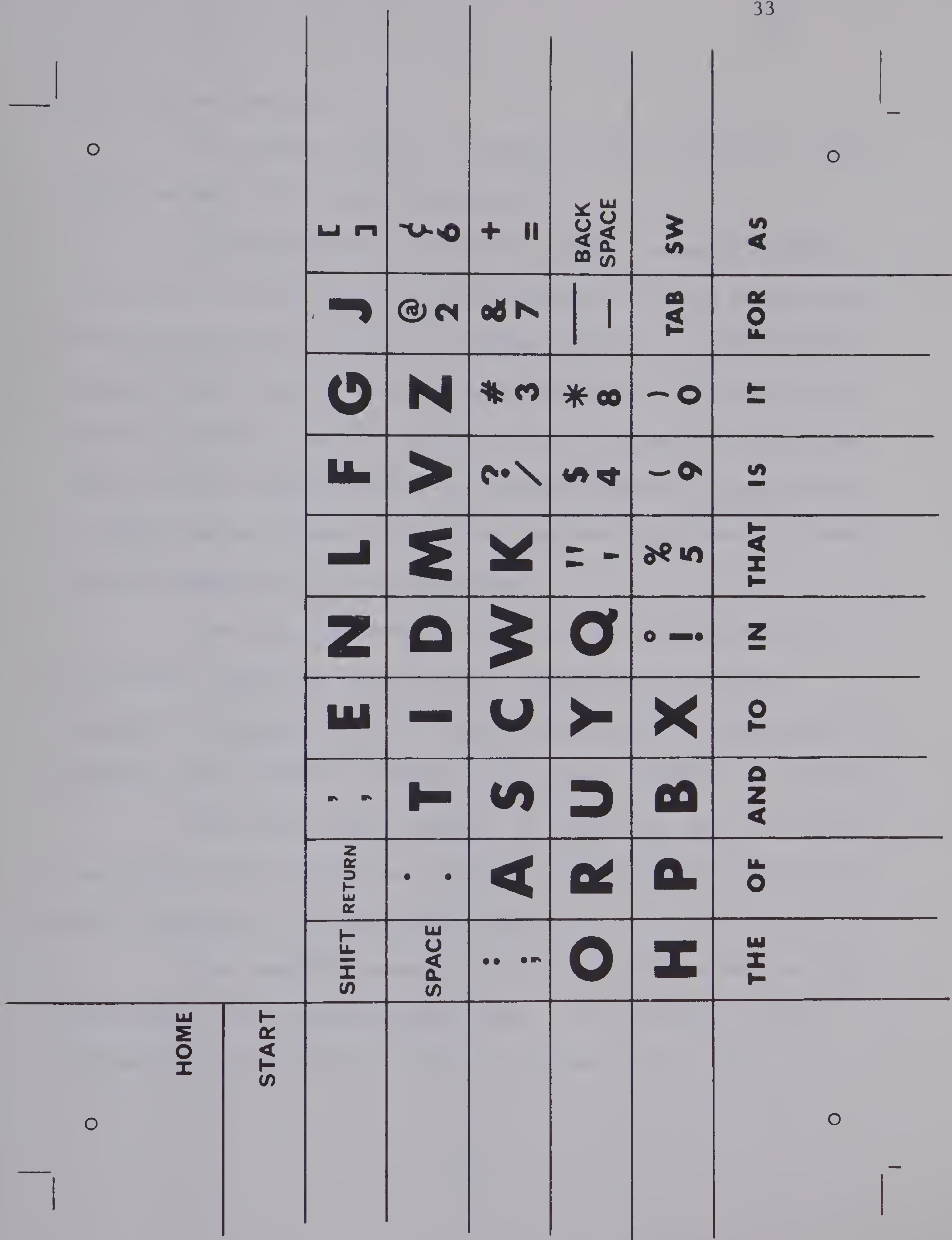


FIG. 2.9 Film Transparency

2.2.2 Output Section

The output section or display section consists of a film display assembly and a lamp PCB assembly.

An exploded view of the film display assembly is shown in Fig. 2.8. It consists of a film transparency placed between two plexiglass plates one of which is opaque acting as a light diffuser. A molded plastic grid is placed between the light diffuser and the lamp PCB assembly. The grid is a section of a standard fluorescent fixture diffuser which provides a convenient method of channelling the light from each lamp into discrete squares to individually back light the characters of the transparency.

The film transparency shown in Fig. 2.9 is made up of 62 9/16 inch squares depicting words, characters and controls available for patient use. The transparency can be easily changed to provide a wide variety of formats with suitable software alterations.

The film display assembly is clamped to the rear of the front plate of the display unit by means of bolts through the front plate to spacers on the lamp PCB assembly.

The lamp PCB assembly consists of a PCB containing lamps, lamp drivers and a lamp alignment plate. The assembly is held together with stand offs and spacers as shown in Fig. 2.8.

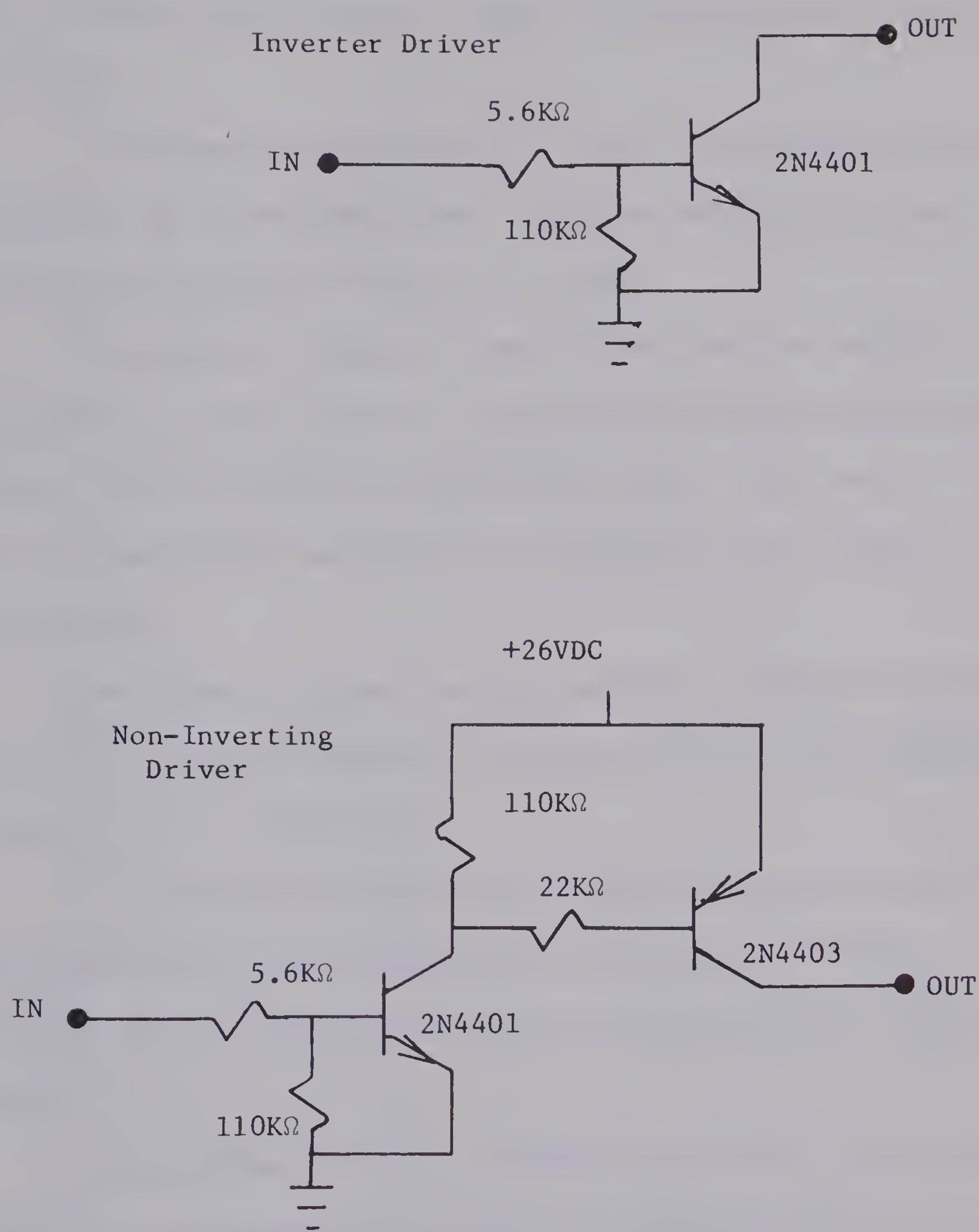


FIG. 2.10 Lamp Driver Circuits

The lamp PCB contains 68 lamps driven by eight row lamp drivers, 11 column lamp drivers, a "shift" lamp driver and a diode matrix decoder.

The lamp's used are G.E. No. 2137D incandescent indicator lamps operating at 28 VDC and 40 ma. They are baseless with bare leads which were soldered directly to the PCB.

To energize a lamp two lamp drivers must be enabled, a non-inverting, or row, driver to supply +26 VDC and an inverting, or column, driver to provide a ground return path. The only exception is the "shift" lamp which is energized from a single inverting driver.

The circuit diagram for the individual drivers is shown in Fig. 2.10. They are designed to be driven from Digital Equipment Corporation (DEC) 'M' Series logic.

The non-inverting drivers, of which there are 8, are a standard saturating driver configuration, providing a driving capability of 100 ma (5Volts input) with a maximum supply voltage of 60 VDC.

The inverter drivers, of which there are 12, are a standard saturating driver design providing a driving capability of 100 ma (5 Volts input) and a maximum voltage output of 60 VDC (0 Volts input).

The arrangement of lamps and lamp drivers is such that

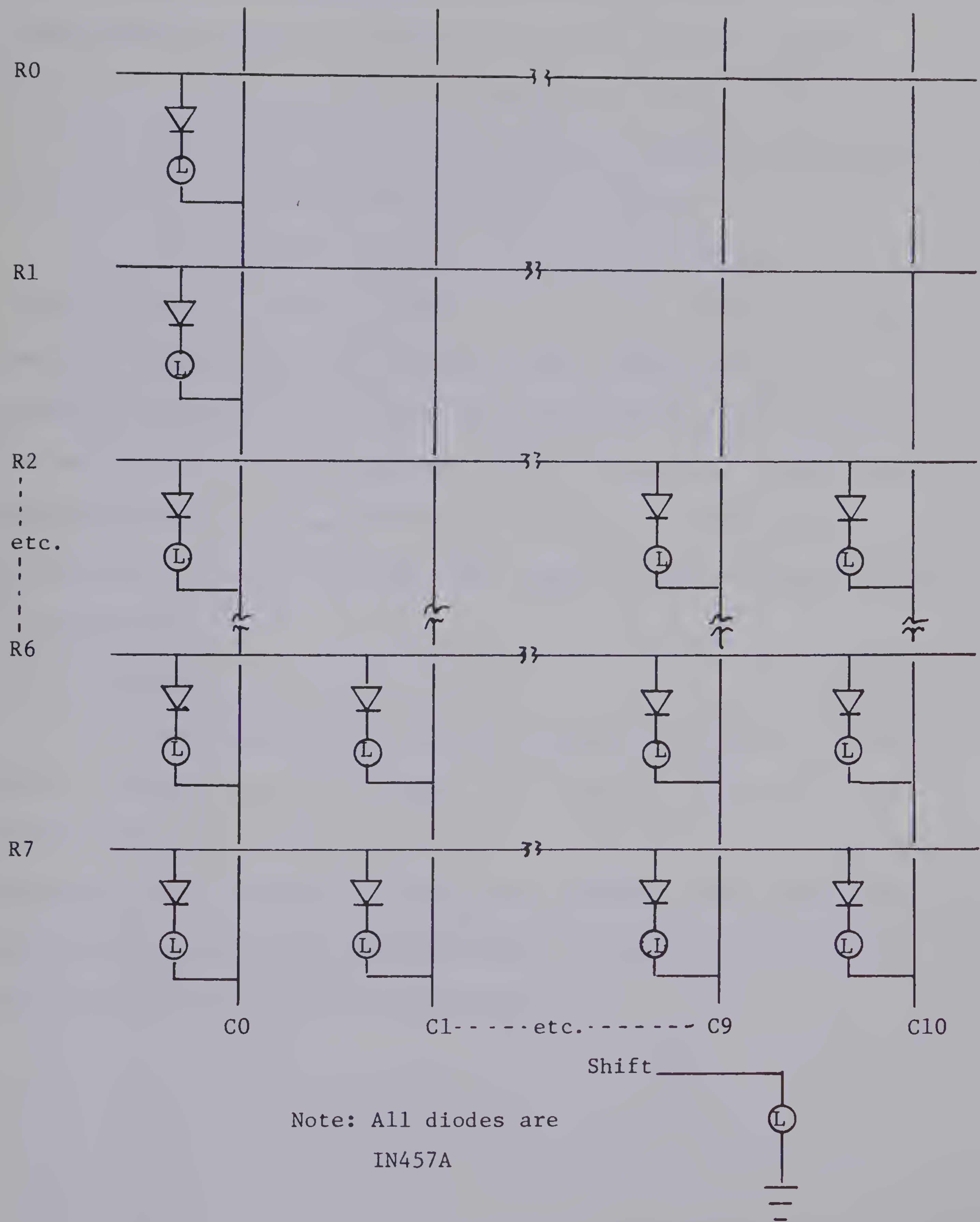


FIG. 2.11 Lamp Diode Decoding Matrix

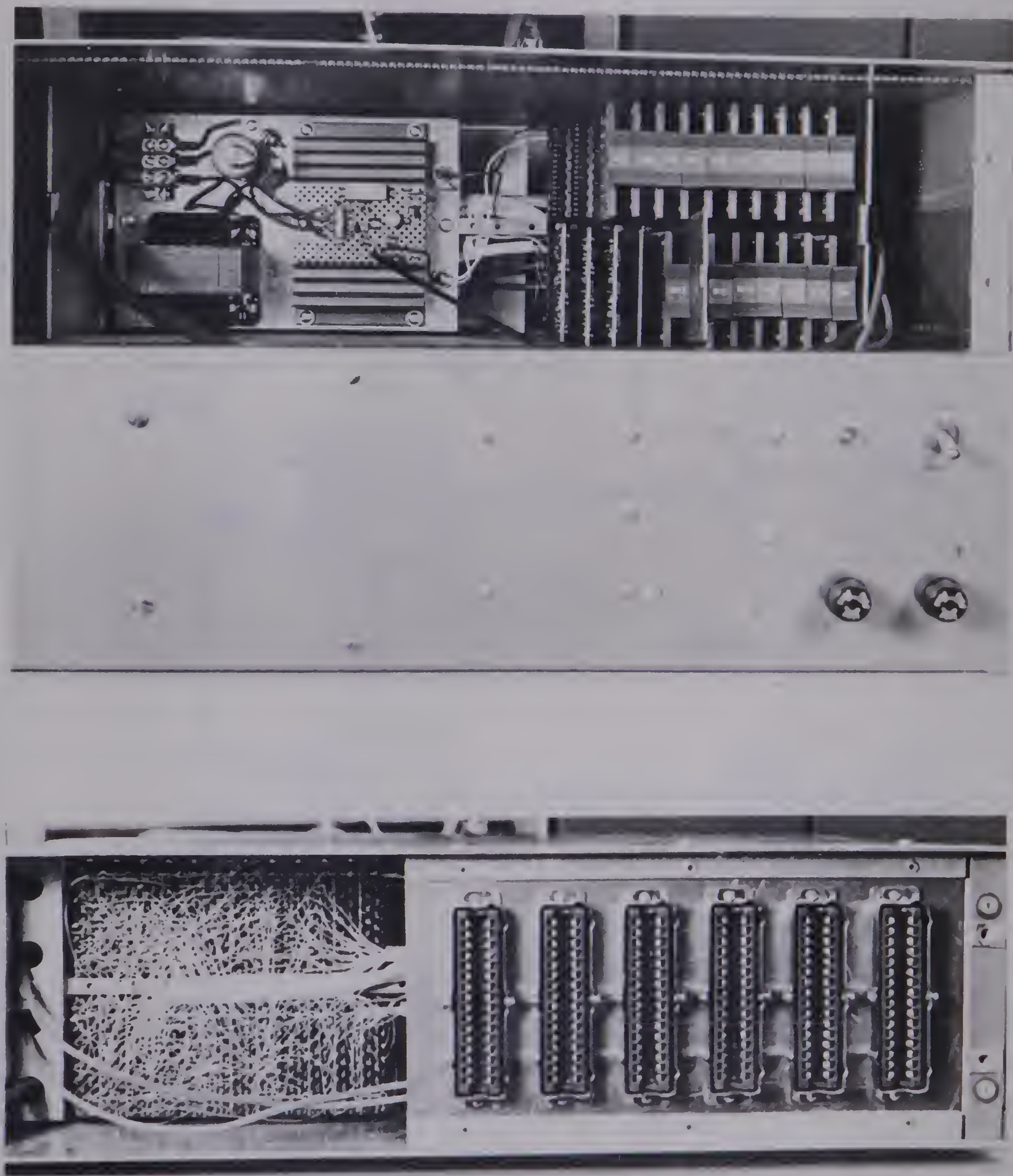
19 control lines can illuminate any one of the 68 lamps. This was accomplished by means of a diode matrix decoder similar to that described in section 2.1.3 for the typewriter actuator. The row and column control lines to illuminate any character corresponds directly to that of the typewriter actuator decoder.

The lamp diode decoder matrix is an X-Y decoder the circuit diagram for which is shown in Fig. 2.11. The circuit uses one IN457A diode associated with each lamp. This provides the necessary decoding such that any combination of one column and one row will only illuminate one lamp. The only exception is the "shift" lamp which has its own control line and driver and which can be illuminated in conjunction with other lamps in order to indicate upper case characters.

2.2.3 Cabling

One main cable interconnecting the display unit with the central interface unit is provided. The cable was hard wired to the display PCB and switches, providing interconnection between both the input and output sections to the central interface unit. Connection with the interface unit is made by means of connector's CN1 and CN4 for patient terminals A and B respectively.

Fig. 2.12 Central Interface Unit



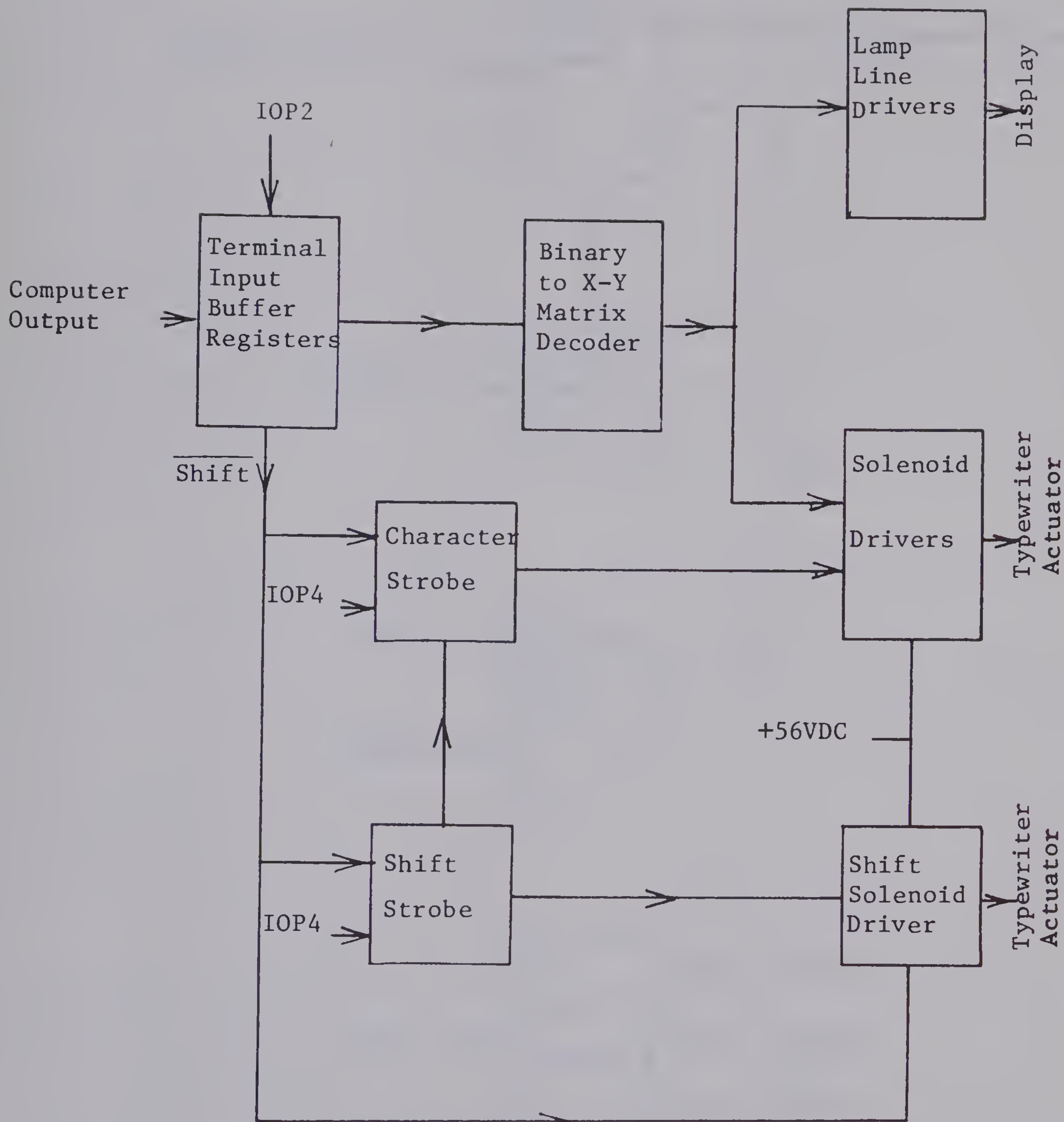


FIG. 2.13 Ouput Interface

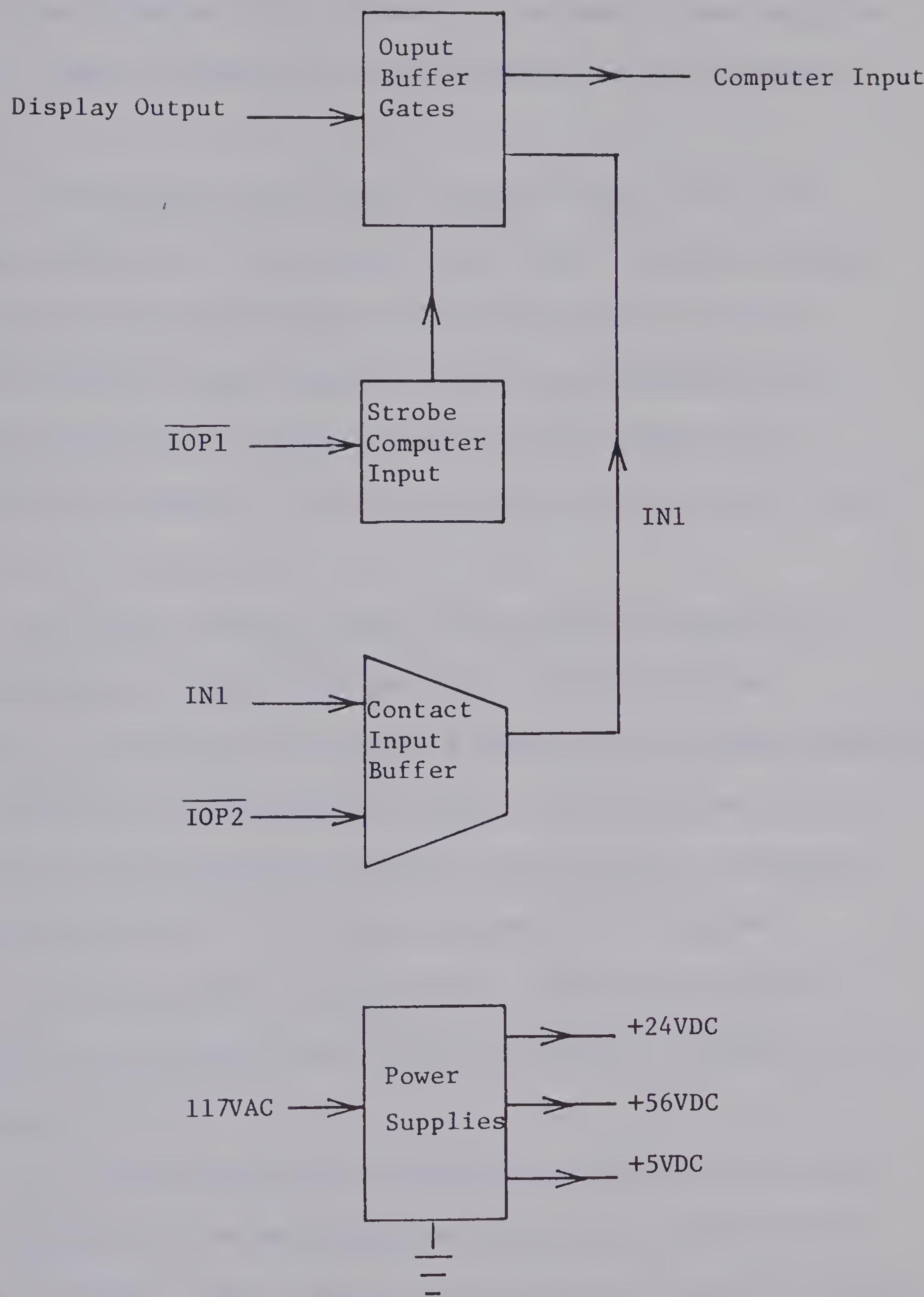


FIG. 2.14 Input Interface

2.3 Central Interface Unit

The interface unit contains all necessary power supplies and computer input and output interface hardware for both patient terminals.

A photograph of the unit is shown in Fig. 2.12. The front panel contains the power supply on/off switch and fuses which control power to the entire system. The rear area of the unit was provided with six cable connectors which were designated CN-1 through CN-6, from left to right. The connectors enable interconnection of the computer, displays and typewriter actuator to the unit.

The logic circuits used in the interface hardware are composed primarily of Digital Equipment Corp (DEC) 'M' Series logic boards. This logic is composed of high speed monolithic integrated circuits employing TTL integrated circuits. The individual modules will not be discussed in detail in this thesis, the reader is referred to reference 9 if further information is required.

Solenoid drivers and time delay PCB's were designed specifically for this project and will be discussed at length in the applicable section.

The output and input interface flow diagrams are shown in Figs. 2.13 and 2.14 respectively, and define the interfacing for one patient terminal. Both terminals use identical interface hardware therefore all discussions in this section apply equally to both terminals.

The interface hardware allows the computer to multiplex control between the two terminals. Associated with each terminal is a computer input/output (IO) device number, terminal A is assigned 35 and terminal B with 36. The device selector of the computer allows the software to generate a series of three pulses, IOP1, IOP2 and IOP4 to be applied to the interface hardware of the desired terminal. These pulses are used to strobe control data into or out of the computer's accumulator. In this way the computer can establish control of a particular terminal over common IO buses.

2.3.1 Output Interface Hardware

Computer control commands are transmitted over computer output buses to terminal input buffer registers. The control word is composed of 6 bits of the 12 bits available from the computer.

Bits 0 and 2 through 8 are used in conjunction with IOP pulses providing the information necessary to perform all required control functions.

The control word is strobed into the terminal input buffer register by means of IOP2. The word is stored there until the next computer output cycle. The output of the register provides a continuous input, to a binary to X-Y matrix decoder, of the control word and its complement. The decoder is composed of two binary to

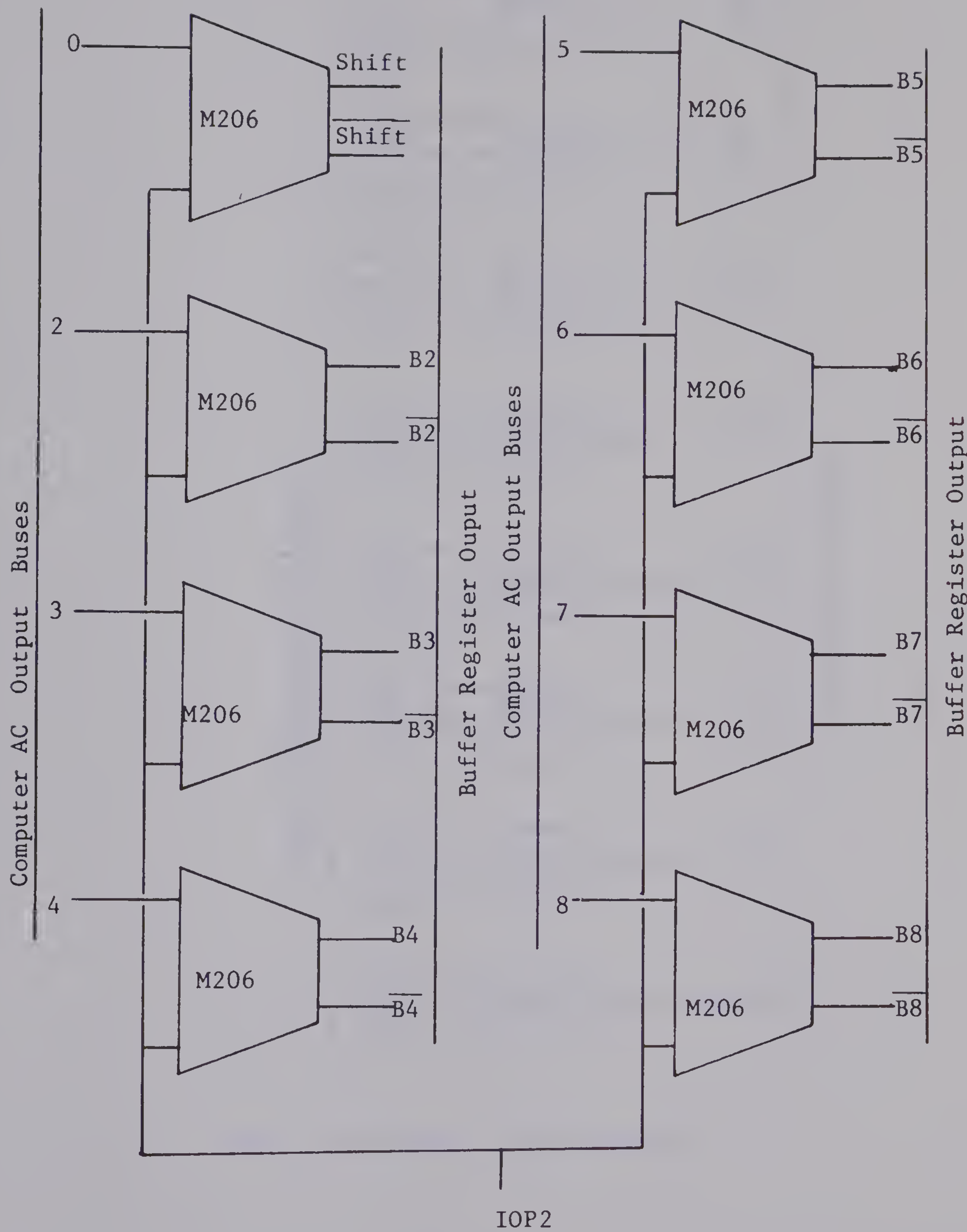


FIG. 2.15 Input Buffer Registers

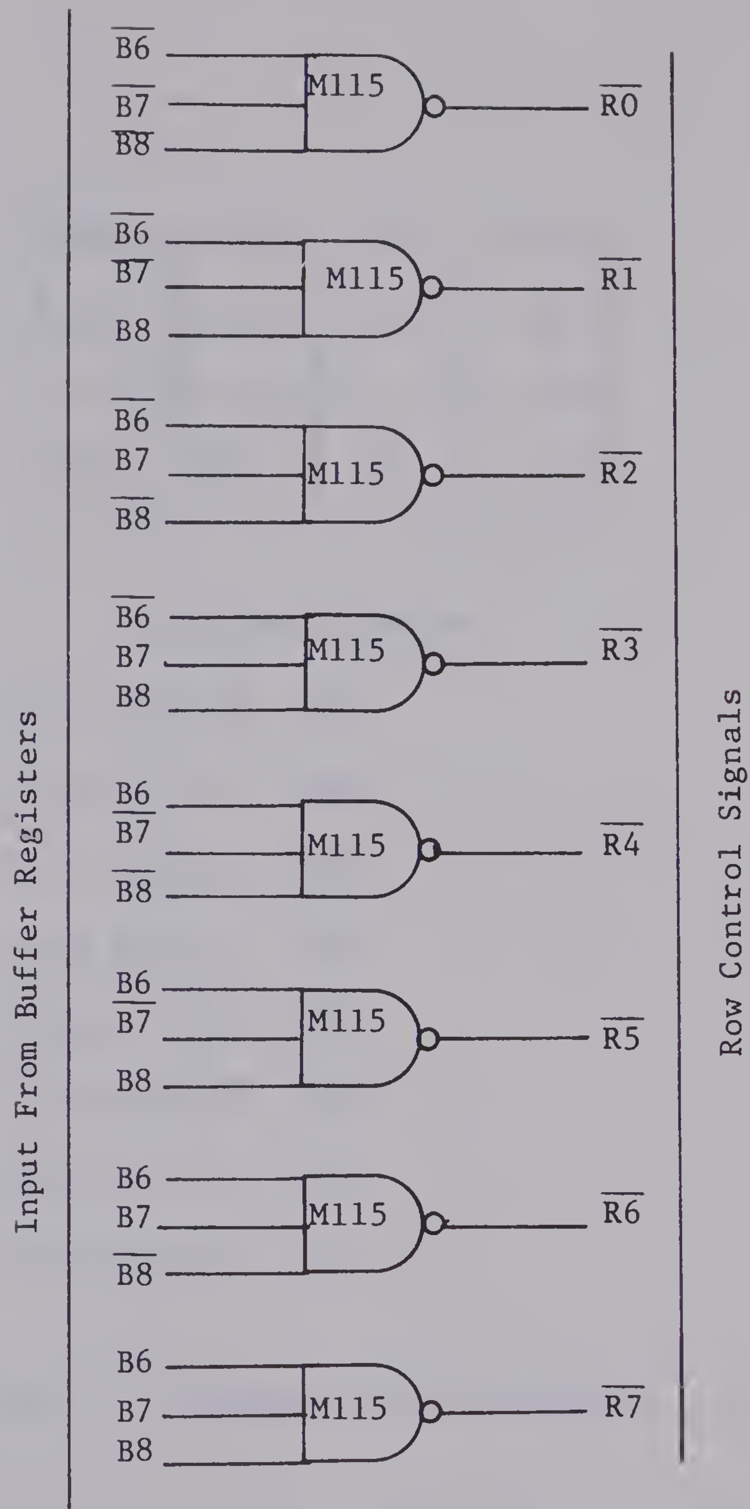


FIG. 2.16 Binary to Row Decoder

B6 B7 / B8	00	01	11	10
0	R0	R2	R6	R4
1	R1	R3	R7	R5

Boolean Formulas

$$R0 = \overline{B6} \cdot \overline{B7} \cdot \overline{B8}$$

$$R1 = \overline{B6} \cdot \overline{B7} \cdot B8$$

$$R2 = \overline{B6} \cdot B7 \cdot \overline{B8}$$

$$R3 = \overline{B6} \cdot B7 \cdot B8$$

$$R4 = B6 \cdot \overline{B7} \cdot \overline{B8}$$

$$R5 = B6 \cdot \overline{B7} \cdot B8$$

$$R6 = B6 \cdot B7 \cdot \overline{B8}$$

$$R7 = B6 \cdot B7 \cdot B8$$

FIG. 2.17 Karnaugh Map Row Decoder

decimal decoders, a row decoder and a column decoder. The decoder outputs drive the lamp drivers and typewriter drivers in parallel. In this way the computer can illuminate any position on the display and type the associated character.

Computer control pulse IOP4 is used to strobe the typewriter driver circuits. Two strobe pulses are generated, one for characters and control keys and the other for the shift solenoid. The strobe circuits determine the timing and duration of key actuation to ensure smooth typewriter operation.

(a) Terminal Input Buffer Register

The logic diagram for the buffer register is shown in Fig. 2.15. It consists of 8 DEC, M206, general purpose flip-flops, one for each bit of the control word. The flip-flops are D type and they are wired with a common input data gate. The data gate is strobed by means of IOP2 to route data to the desired terminal register.

(b) Binary to X-Y Matrix Decoder

The binary to X-Y Matrix decoder decodes the buffer register bits 2 through 8 to establish the row and column (X and Y) control signals to illuminate the display and actuate the typewriter solenoids.

The desired row is derived from bits 6,7 and 8 using a binary to decimal converter composed of eight 3 input nand gates (DEC type M115) as shown in Fig. 2.16. A Karnaugh Map analysis of this decoder is shown in Fig. 2.17.

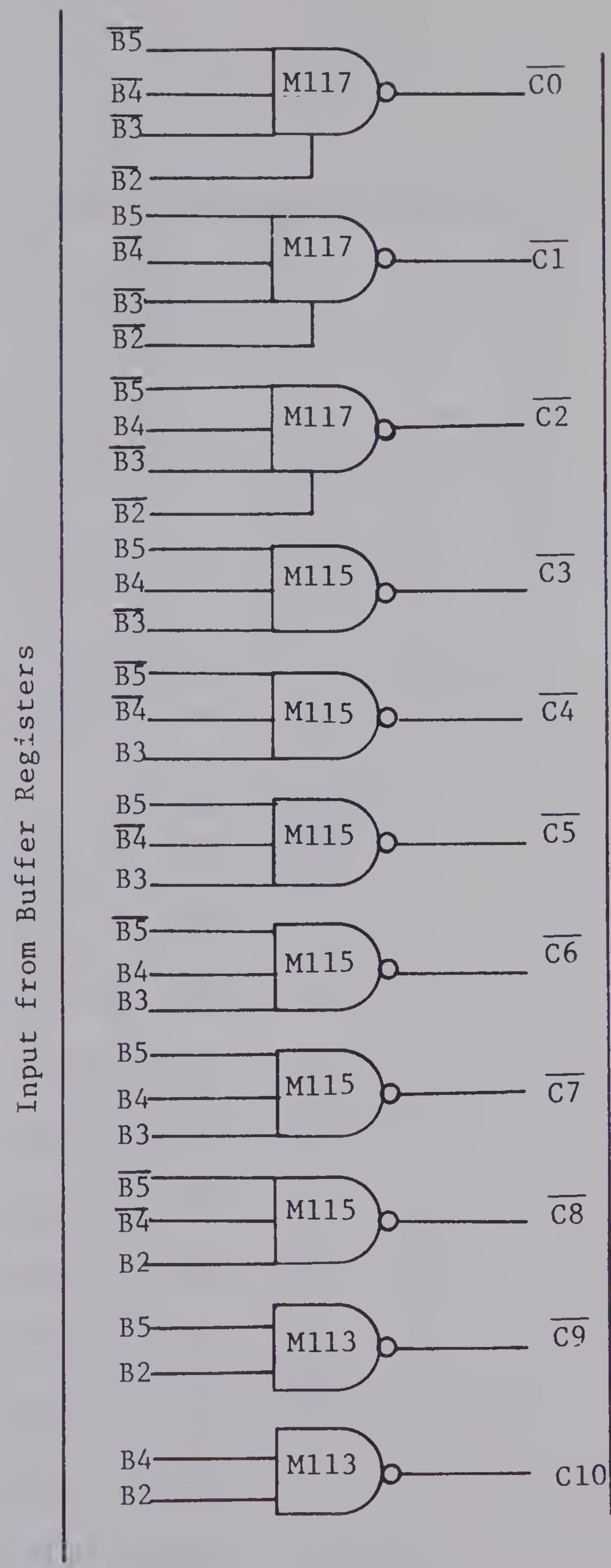


FIG. 2.18 Binary to Column Decoder

B4 B5 \ B2 B3	00	01	11	10
00	C0	C4	-	C8
01	C1	C5	-	C9
11	C3	C7	-	-
10	C2	C6	-	C10

Boolean Formulas

$C0 = \overline{B2} \cdot \overline{B3} \cdot \overline{B4} \cdot \overline{B5}$

$CI = \overline{B2} \cdot \overline{B3} \cdot \overline{B4} \cdot B5$

$C2 = \overline{B2} \cdot \overline{B3} \cdot B4 \cdot \overline{B5}$

$C3 = \overline{B3} \cdot B4 \cdot B5$

$C4 = B3 \cdot \overline{B4} \cdot \overline{B5}$

$C5 = B3 \cdot \overline{B4} \cdot B5$

$C6 = B3 \cdot B4 \cdot \overline{B5}$

$C7 = B3 \cdot B4 \cdot B5$

$C8 = B2 \cdot \overline{B4} \cdot \overline{B5}$

$C9 = B2 \cdot B5$

$C10 = B2 \cdot B4$

FIG. 2.19 Karnaugh Map Column Decoder

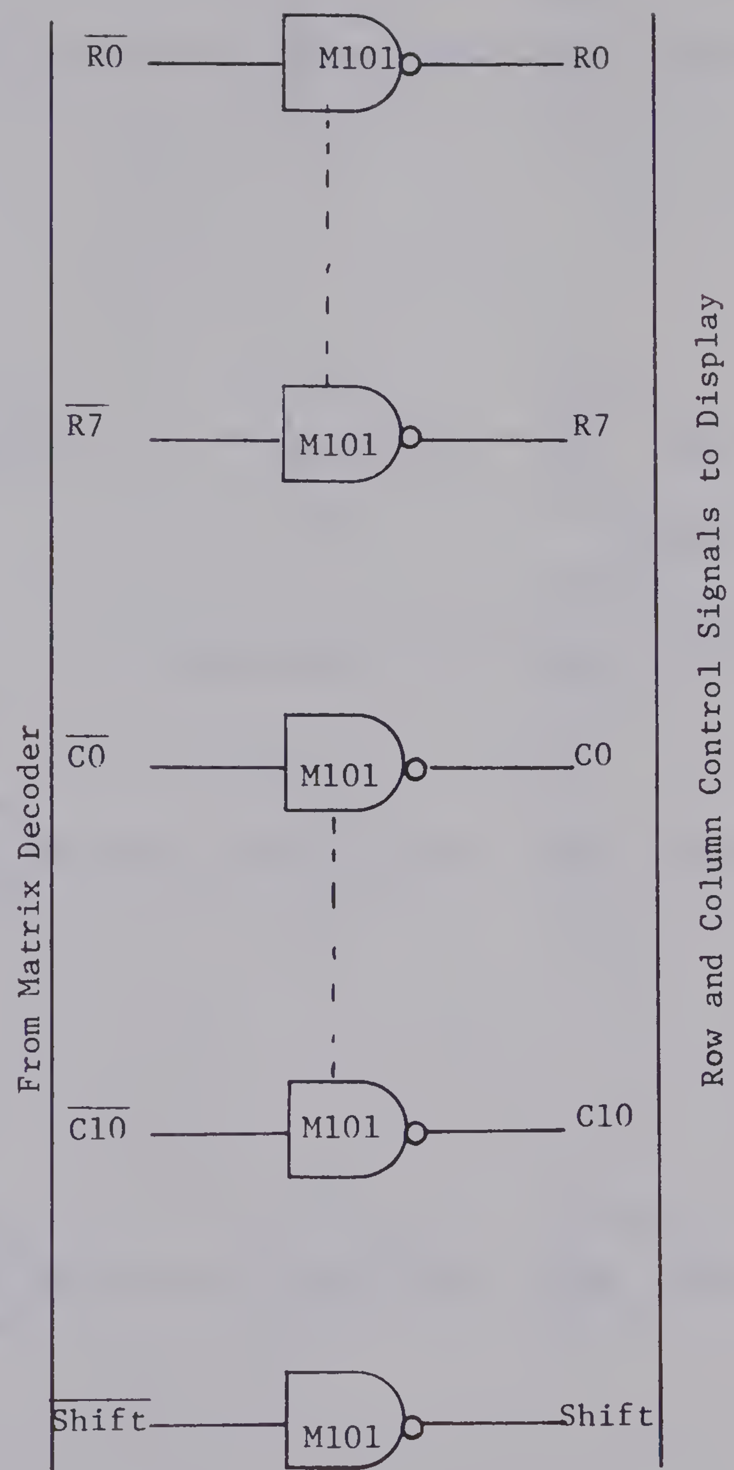


Fig. 2.20 Lamp Line Drivers

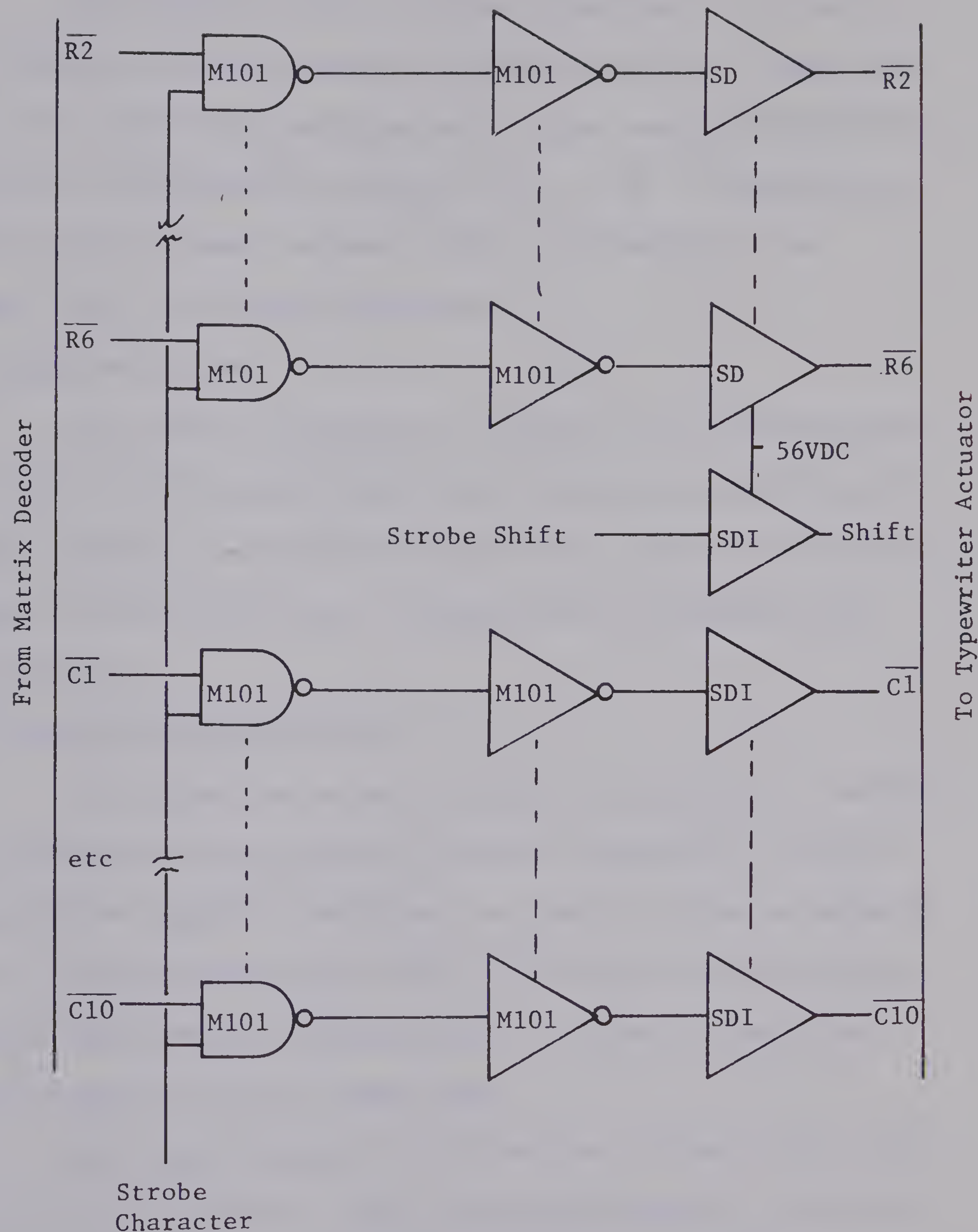


FIG. 2.21 Gated Solenoid Drivers

to eliminate redundancy.

The desired column is derived from bits 2, 3, 4 and 5 using a binary to decimal converter composed of three 4 input nand gates, six 3 input nand gates and two 2 input nand gates (DEC, M117, M115 and M113 respectively) as shown in Fig. 2.18. A Karnaugh Map analysis of this decoder is shown in Fig. 2.19 indicating the necessary logic to eliminate redundancy.

(c) Lamp Line Drivers

The lamp line drivers are a series of 19 inverter gates used to invert the row and column control signals obtained from the decoders, providing the necessary line driving capability to activate the lamps of the display unit. The gates used are DEC M101 data transfer gates.

(d) Typewriter Solenoid Drivers

The typewriter solenoid drivers are connected in parallel to the display lamp line drivers on rows R2 through R6, columns C1 through C10 and "shift". The drivers are gated to allow strobing by means of a common control bus, with a type pulse of fixed duration. The strobe pulse enables separate control of display scanning and typewriter operation from a common input.

The logic diagram for this section is shown in Fig. 2.21. The row drivers consist of 2 input nand gates followed by inverters driving non-inverting solenoid drivers. The column drivers consist of 2 input nand gates, followed by inverters driving inverting solenoid drivers. The "shift" solenoid is activated by a single non-

FIG. 2.22 Inverting Solenoid Driver

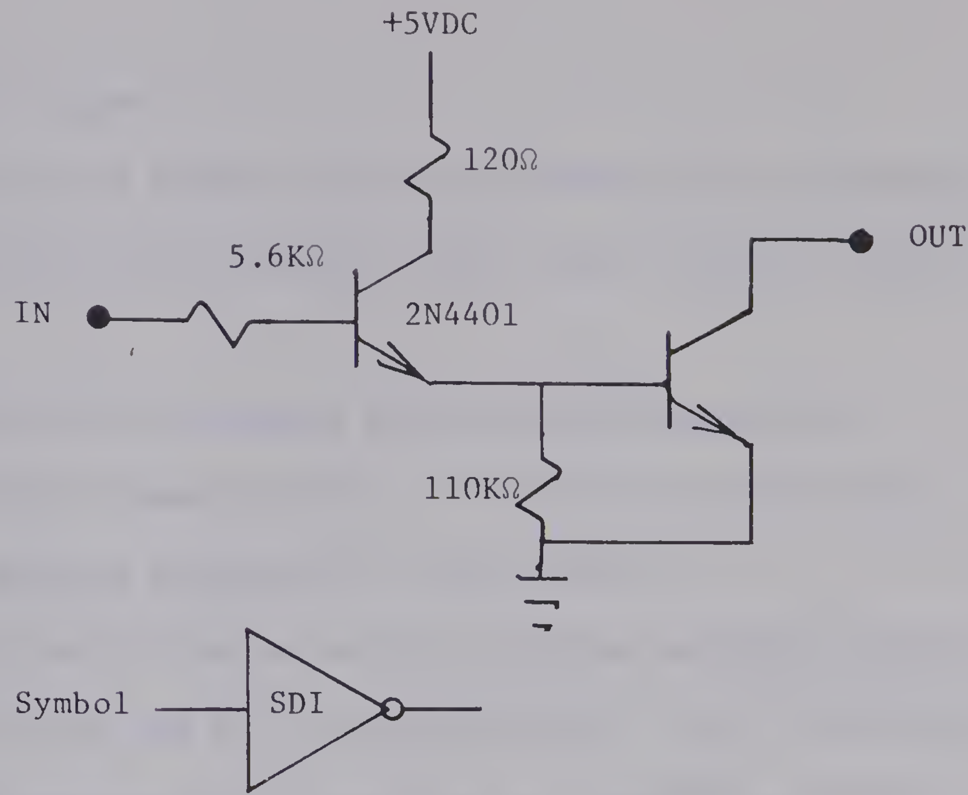
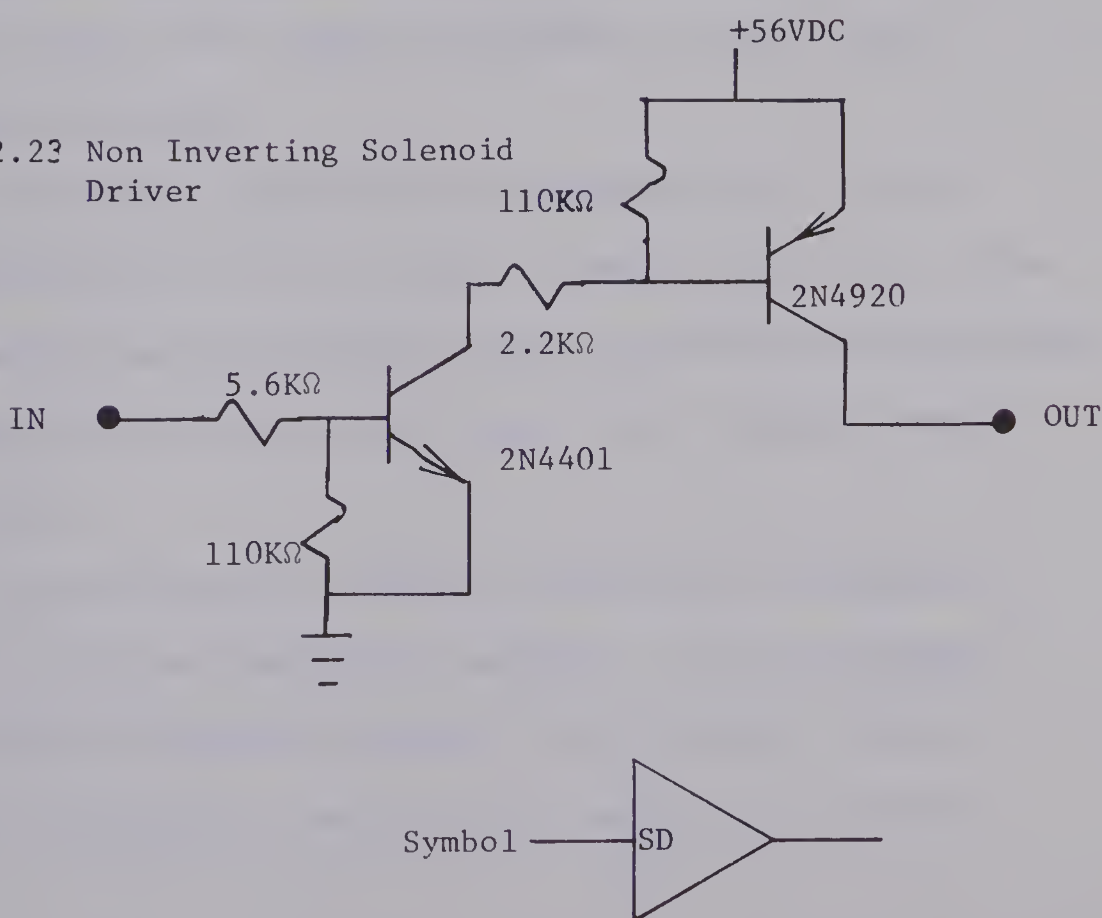


FIG. 2.23 Non Inverting Solenoid Driver



inverting solenoid driver.

The row and column drivers are controlled by one common character strobe pulse and the shift driver from a separate "shift" strobe pulse.

The 2-input nand gates and the inverter gates are composed of DEC data transfer modules. The solenoid drivers are special purpose modules designed for this project.

The circuit diagram for the inverting solenoid drivers, of which there are six per PCB, is shown in Fig. 2.22. The design is a saturating inverter driver providing an intermittent driving capability of 1 amp (5 Volt input) and maximum voltage output (0 Volts input) of 80 VDC.

The circuit diagram for the non-inverting solenoid driver, of which there are six per PCB, is shown in Fig. 2.23. The design is a saturating non-inverting driver providing an intermittent driving capability of 1 amp (5 Volt input) with a maximum supply voltage of 80 VDC.

The two drivers are used in conjunction to energize a solenoid. The non-inverting driver energizes a row of solenoids associated with the typewriter actuator, with +56 VDC. The inverting driver establishes a ground return path for the desired

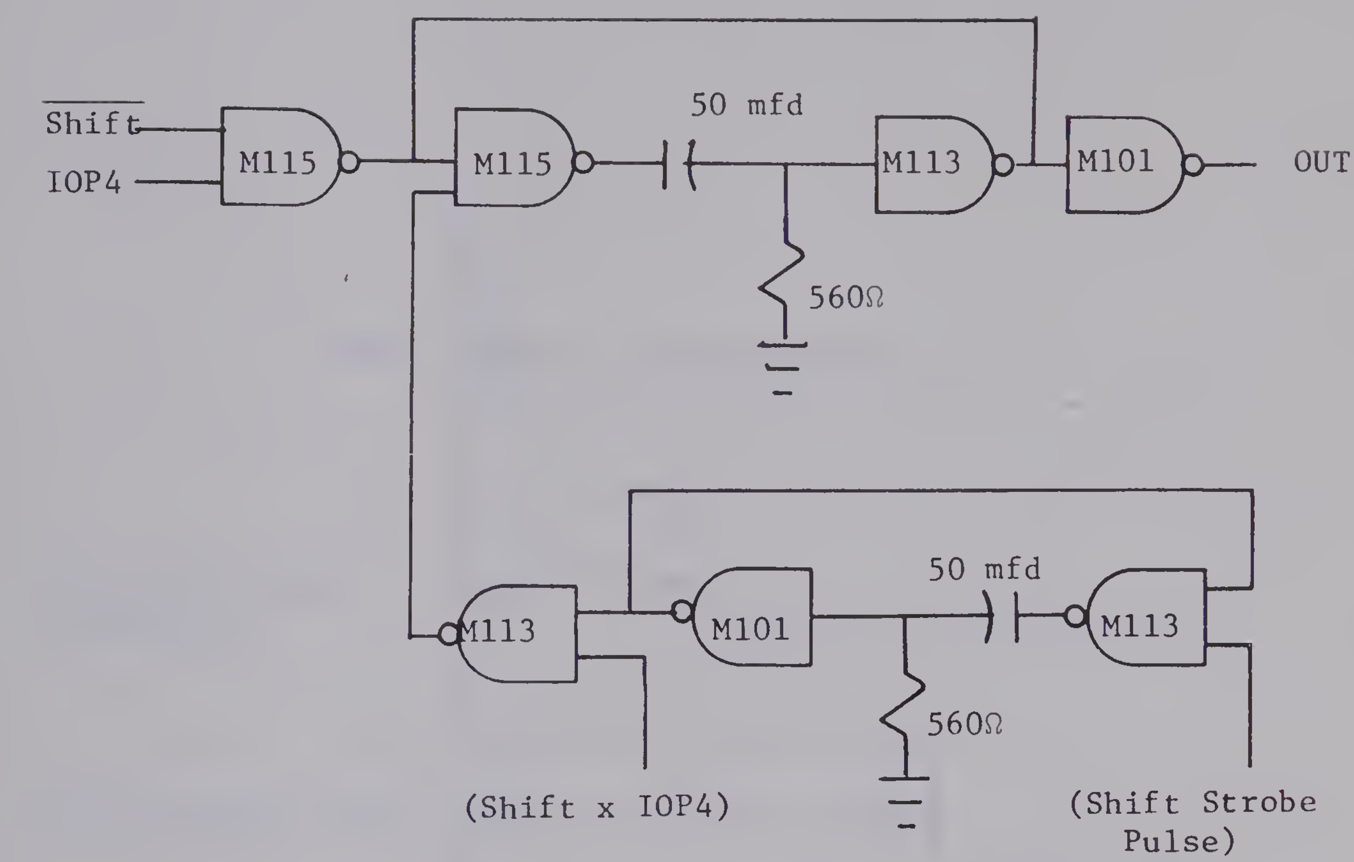


FIG. 2.24 Character Strobe Pulse

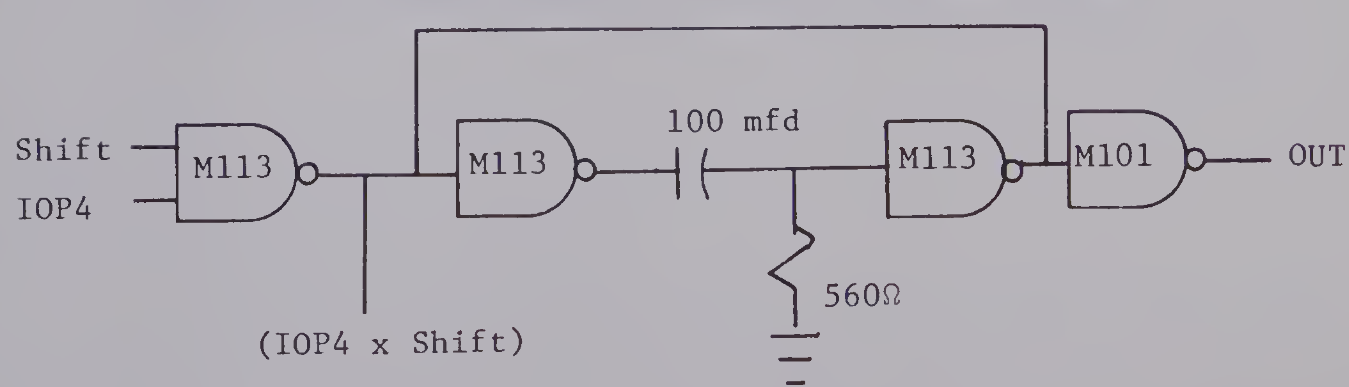


FIG. 2.25 Shift Strobe Pulse

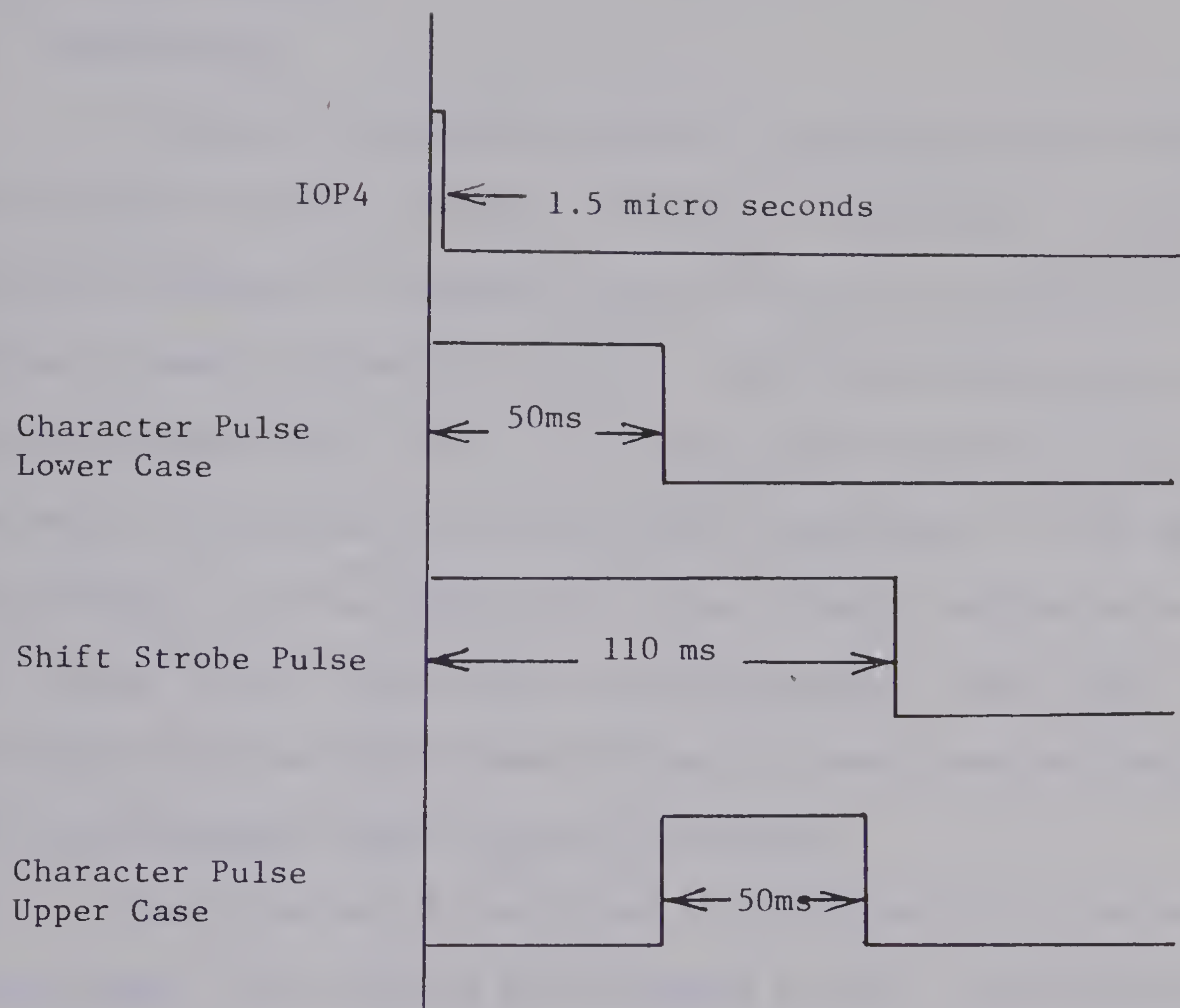


FIG. 2.26 Timing Diagram Typewriter Strobe Pulses

solenoid in the row. This method of energizing the solenoids by means of an X-Y matrix system minimizes the number of drivers and control lines required.

(e) Strobe Pulses

The logic required to generate the character and "shift" strobe pulses is shown in Figs. 2.24 and 2.25 respectively. Both pulses are initiated by computer pulse IOP4 and are of sufficient duration to ensure solenoid actuation. The timing diagrams for these pulses are shown in Fig. 2.26. If a lower case character is to be typed only a 50 ms character strobe pulse is generated. If an upper case character is to be typed both a 50 ms character strobe pulse and a 110 ms "shift" strobe pulse will be generated. The "shift" pulse will preceed the character pulse by 50 ms to ensure that the "shift" key is fully engaged before typing the character.

The duration of the pulses is determined by the monostable circuits shown. The circuits are triggered by IOP4, conditional on whether "shift" is requested, and extends the pulse for a duration determined by the time constant of the R-C networks. The output of the monostable is fed back to its input to maintain the pulse until the capacitor is discharged.

2.3.2 Input Interface Hardware

Terminal mode settings and switch input signals are transmitted to the computer over computer accumulator input buses. The input word is composed of 7 bits of the 12 input buses of the accumulator.

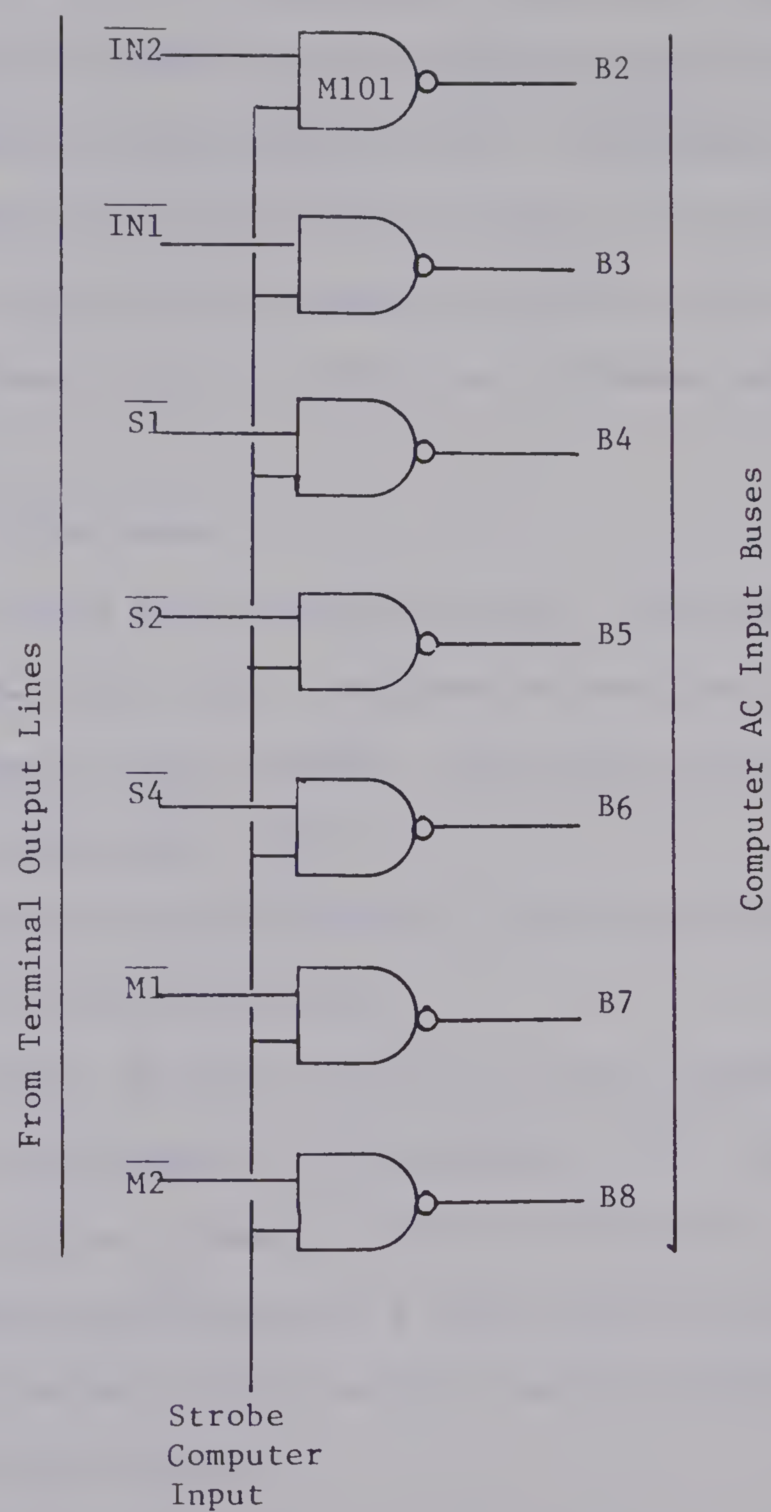


FIG. 2.27 Output Buffer Gates

Bits 2 through 8 provide all the information necessary as to the mode selected, scan rate selected and state of the input device. This input word is strobed into the computer's accumulator by means of terminal output buffer gates and IOP1. All terminal outputs are in parallel and selection of the desired terminal is accomplished by generating the IOP1 pulse for the terminal in question. The word is stored as a software control variable and is updated each computer control cycle.

(a) Terminal Output Buffer Gates

The output buffer gates consist of seven 2 input nand gates (DEC M115) and are shown in Fig. 2.27. One input of each gate is common and used for computer input strobing. The other input is connected to a terminal data output line.

Output lines IN1 and IN2 are switch inputs 1 and 2 respectively, this project only utilizes IN1.

Output lines S1, S2 and S4 provide an octal representation of the terminals "scan rate" selector switch position. The three signals are strobed through as accumulator bits B4, B5 and B6.

Output lines M1 and M2 provide a binary coded representation of the terminals "mode" selector switch position and are strobed through as accumulator bits B7 and B8.

The terminal output lines are obtained from switch closures

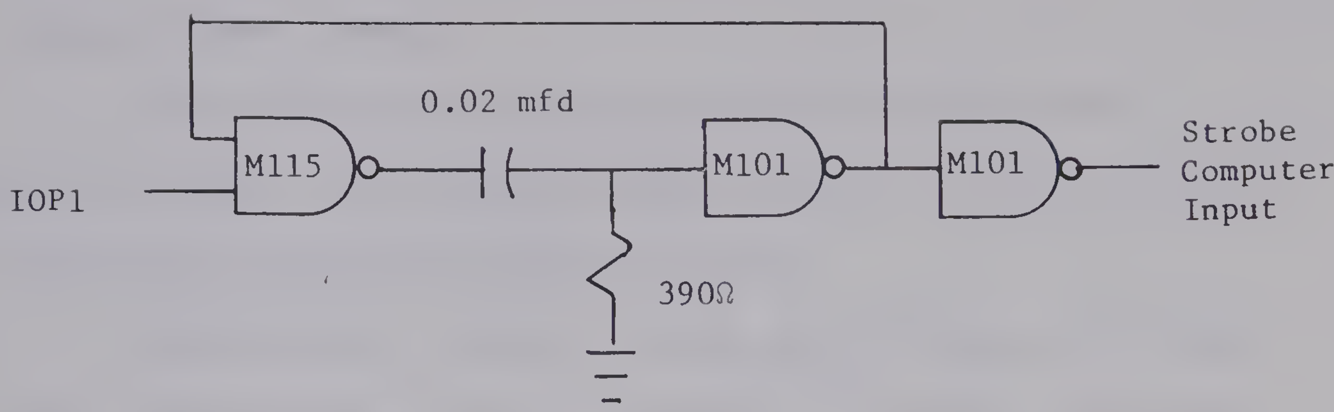


FIG. 2.28 Strobe Computer Input Pulse

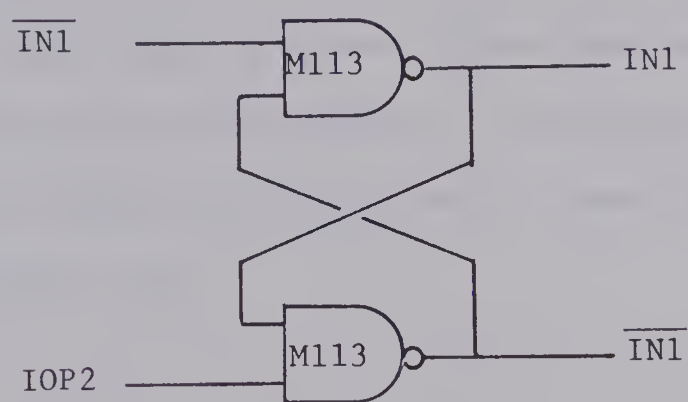


FIG. 2.29 Contact Input Buffer

such that a "1" is an open circuit and a "0" is ground.

(b) Computer Input Strobe

The strobe pulse to enable the output buffers is an extended IOP1 pulse. The circuit is shown in Fig. 2.28 and it consists of a monostable followed by an inverter driver.

The monostable circuit extends the IOP1 pulse for 6.5 ms to provide sufficient time for the computer to accept the input data after generating the IOP1 pulse. The pulse duration is determined by the time constant of the R-C network. The IOP1 pulse initiates the strobe pulse which is fed back to maintain the pulse until the capacitor is discharged.

(c) Contact Input Buffer

The input buffer consists of two 2 input nand gates (DEC M113) wired in a flip-flop configuration. The circuit is set by $\overline{\text{IN1}}$ and reset by IOP2 thereby eliminating switch input changes during a data input transfer cycle.

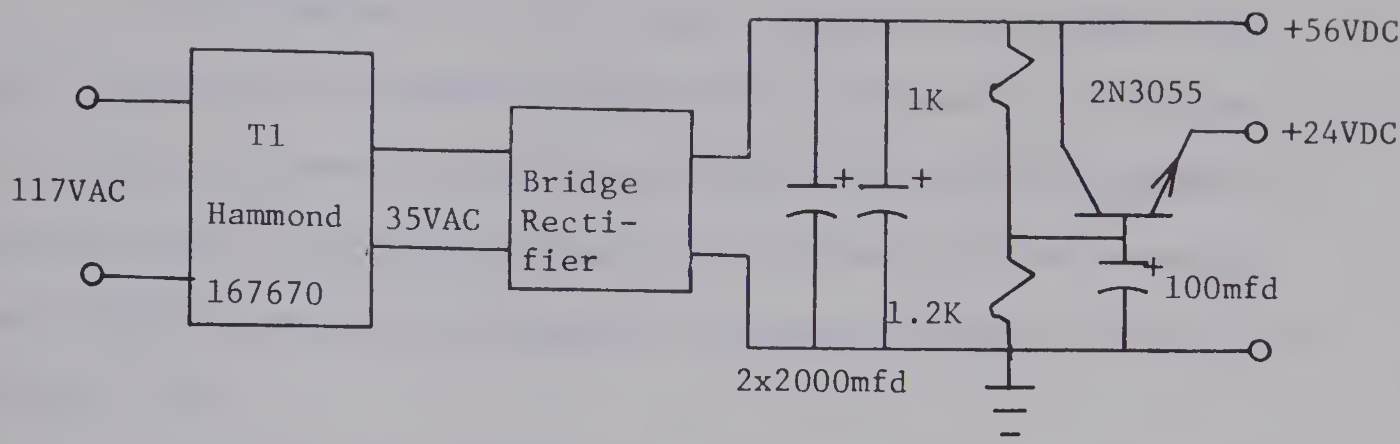


FIG. 2.30 Power Supply P1

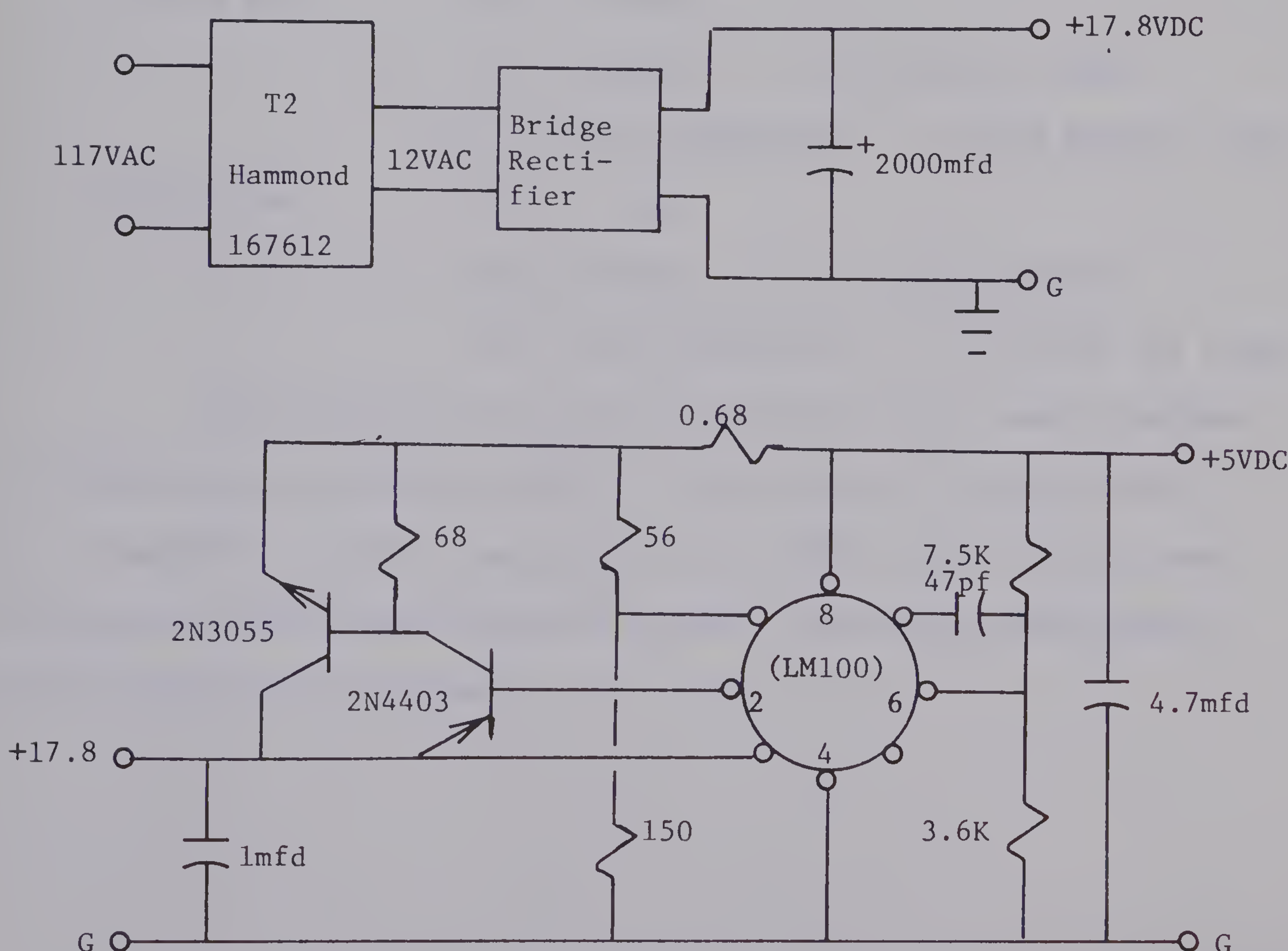


FIG. 2.31 Power Supply P2

2.4 Power Supplies

The hardware for this system requires three supply voltages, 56 volts to energize the solenoids of the typewriter actuator, 26 volts for the lamps of the display units and 5 volts to supply the interface logic. The three voltages are obtained from two separate power supplies, the circuit diagrams for which are shown in Fig. 2.30 and Fig. 2.31.

Power supply one is a simple unregulated supply providing 56 volts and 26 volts. The specifications are as follows

56 volts bus	(a)	1 amp
	(b)	Ripple - 2 volts peak to peak
	(c)	Load Regulation - for 56 VDC and 1 amp
26 volts bus	(a)	1 amp
	(b)	Ripple - 1 volt peak to peak
	(c)	Load Regulation - for 26 VDC and 1 amp

Power supply two is a fully regulated 5 volt supply designed to fulfill the requirements of DEC 'M' series logic. The supply uses an IC regulator, National Semiconductor Co., LM100, with a 2N3055 power transistor as the series regulating element. The measured performance of this circuit is as follows

(a)	Voltage	-	5.2 VDC
(b)	Current	-	Limited at 2 amps
(c)	Short Circuit Current	-	0.5 amps
(d)	Ripple	-	Less than 1 mv peak to peak
(e)	Load Regulation	-	0.1% for 5.2 VDC and 2 amps
(f)	Line Regulation	-	0.1% for 117 VAC ± 10%

These power supplies are somewhat excessive for the existing hardware as shown by the following load measurements, which were made with one patient terminal in use.

TABLE 2.4

Power Supply Requirements

Voltage	Current (at "Home" position)	Current (maximum)
5.2	0.57 amps	0.58 amps
26	0.42 amps	80 ma
56	0	1.1 amps

These test results indicate the power supply penalty for each additional patient terminal added to the system.

CHAPTER 3

SYSTEM SOFTWARE

The systems program for this project is designated "S:CTSH" which stands for "Computerized Typing Sytem for the Handicapped". The program provides the necessary software to simultaneously control two patient terminals using time sharing techniques based on real time clock pulses.

The program, which appears in Appendix 'A', was written in ASCII (USA Standard Code for Information Interchange) code for use on a Digital Equipment Corporation (DEC) PDP-8 general purpose computer. The ASCII symbolic source program was assembled into a binary object program using the DEC PAL-D Symbolic Assembler program[3]. The complete binary object program occupies 1523_8 memory word locations, from page 0 through to page 7.

The program can be easily extended to incorporate control of additional patient terminals. The major software additions would be new subroutines BEGIN, LOAD, STORE, a set of control variables and minor changes to the central control program. This would constitute an additional 132_8 words for each terminal. On this basis the PDP-8 controls a total of 42 terminals, using pages 0 through 36, with page 37 reserved for the system monitor[3] program.

The general flow diagram for the system program is shown

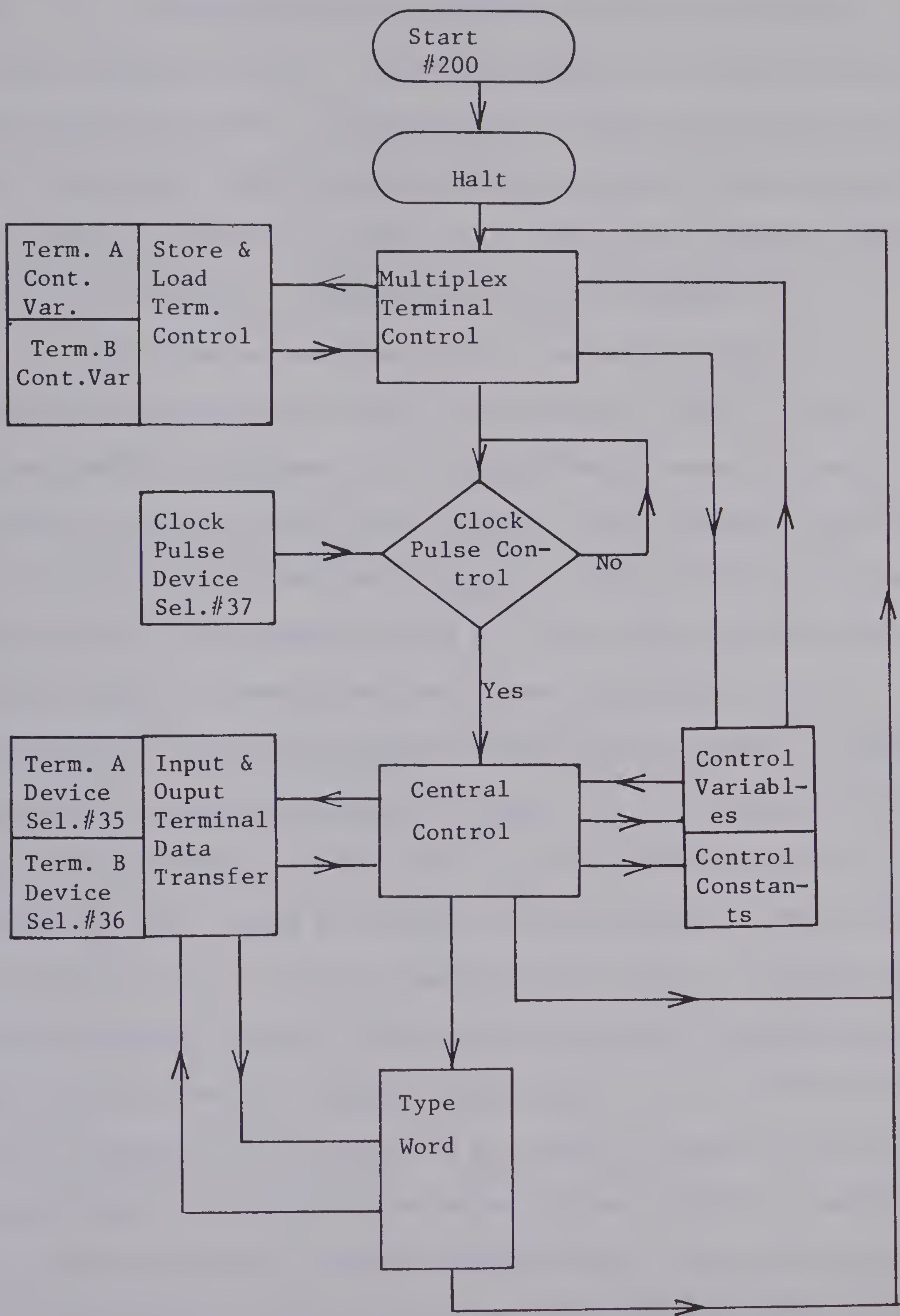


FIG. 3.1 General Program Flow

in Fig. 3.1. It segregates the program into four major sections, "multiplex terminal control". "clock pulse control", "central control program" and "type word". No attempt will be made to describe every "micro" instruction, the discussion will be in general terms relative to the program sections. The sections are composed of groups of subroutines, the programs for which can be found in Appendix 'A'.

The program, once loaded into core memory, halts at address 200 and proceeds only after the "continue" switch on the computer console is depressed. The program then proceeds to transfer the control variables of the first terminal to be processed into those of the "central control program". Once the initial conditions of the terminal are set the program proceeds to "clock pulse control" which halts the program to advance only on the next clock pulse. Each pass through the "clock pulse control" subroutine is counted to establish a time base for terminal scanning and typing. The "central control program" then performs a control cycle on the terminal in control, advancing the scan, typing a character or processing new terminal input data. When the control cycle is complete the program is returned to the "multiplex terminal control" section where the control variables are stored, those of the next terminal to be controlled are loaded and the process is repeated. If a word is to be typed the request is stored as a control "flag" in the control variables for the terminal in question. To type the complete word 6 passes through section "type word" are required. each pass is a control cycle from the "central control program", through "type word" and returning to the multiplex section.

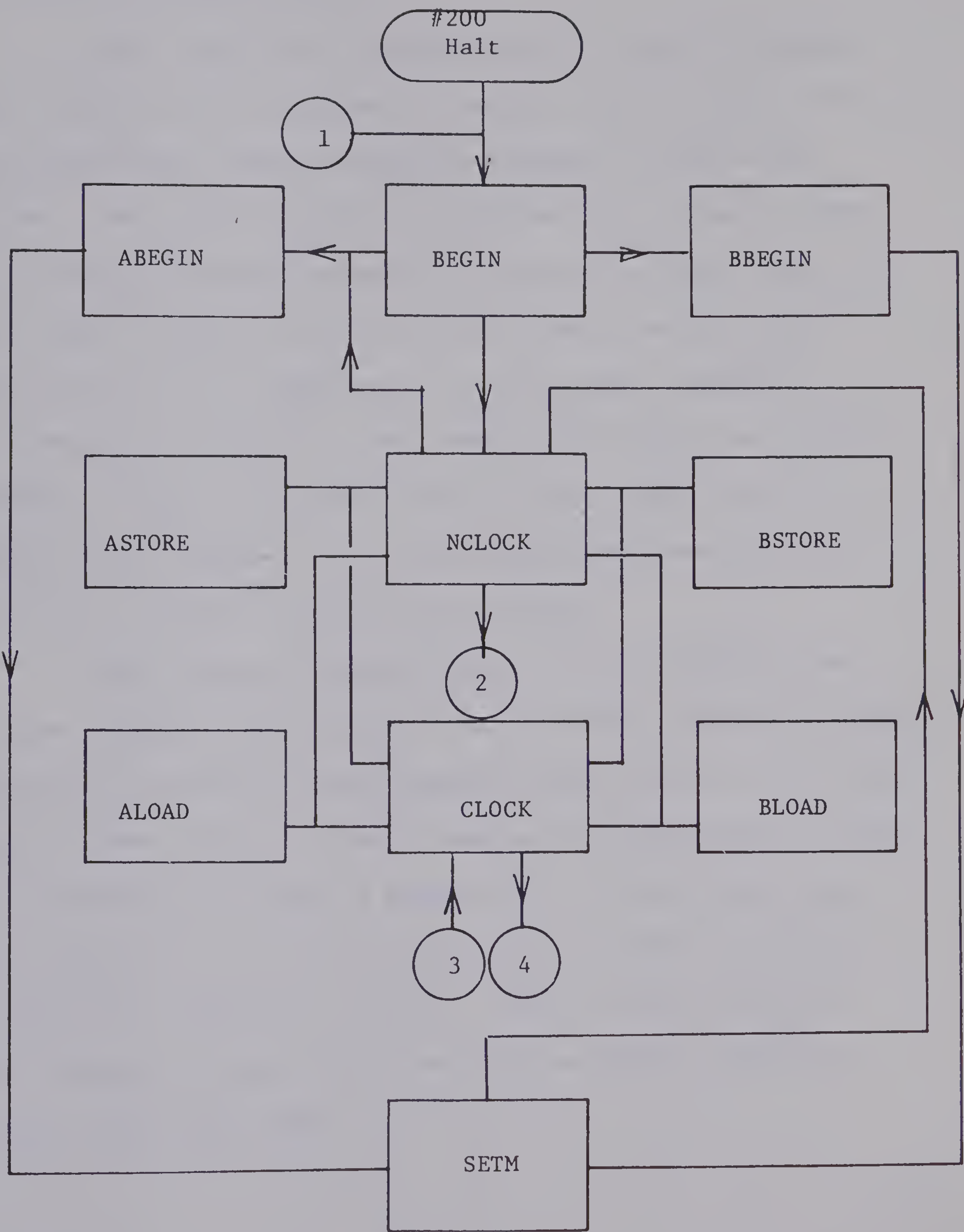


FIG. 3.2 Multiplex Terminal Control

3.1 Multiplex Terminal Control

Each terminal has associated with it a set of software control variables which provides the "central control program" with all the past and present state information necessary to make control decisions. The control variables of terminal A are stored as words 0400_8 to 0426_8 and those of terminal B are stored as words 0427_8 to 0456_8. These variables are not accessed directly but are transferred in and out of a common set of control words, depending on which terminal is in control. The common control words are located in registers 0020_8 to 0046_8 and at any one time contain only the variables of the terminal in control, transferring them in at the beginning of a control cycle and out at the end.

This process of transferring control variables in and out of the "central control program" and alternately with each terminal provides the basic means of multiplexing terminal control. The flow diagram as shown in Fig. 3.2 and is made up of 10 subroutines the programs for which can be found in Appendix 'A'. A pass through this program can start at one of two places, BEGIN or CLOCK depending on the state of the terminal in control. Output to the"clock pulse control program" can come from either CLOCK or NCLOCK depending on the type of pass being made.

The "multiplex terminal control program" will start at BEGIN when the systems program is first initiated or if the terminal in control is to be reset. BEGIN detects which terminal is being controlled. If terminal A is in control the program advances to ABEGIN which reset all control variables of the terminal and proceeds to SETM. If terminal B is in control the program advances to BBEGIN which resets all control variables of the terminal and proceeds to SETM. SETM establishes the mode setting of the terminal in control and advances the program to NCLOCK. If neither terminal is in control, which occurs only when the program is first initiated, the program proceeds directly to NCLOCK which sets terminal A into control and advances to ABEGIN to make a second pass through to NCLOCK.

The subroutine NCLOCK stores the control variables of the terminal in control and loads those of the new terminal to be controlled by means of subroutines ALOAD, BLOAD, ASTORE and BSTORE. The control variables of the new terminal in control are checked to set the scan rate requested and the program is advanced to "clock pulse control". NCLOCK processes the multiplexing of control variables only when a terminal is to be reset or on the first pass of the program.

The subroutine CLOCK performs the same function as NCLOCK for all control cycle passes other than those requesting terminal reset. In addition to multiplexing terminal control variables,

CLOCK also deposits the terminals present data input state into a past state register to provide the "central control program" with a means of determining any change, and its direction. After the control variables of the terminal to be controlled are set the program advances to the "clock pulse control" program.

The "multiplex terminal control" program sets the terminal to be processed by the "central control program" by establishing control "flags". The control "flag" for terminal A is TA and for terminal B is TB. The "central control program" monitors these "flags" to determine which terminal is set. This information is used by the control program to make the correct input and output data transfers at various points in the program.

3.2 Clock Pulse Control

The "clock pulse control" program provides a real time basis for the scanning and typing processes. The flow diagram for this program is shown in Fig. 3.3 and is composed of subroutine SKIP and an input transfer of the clock pulses via device selector number 37.

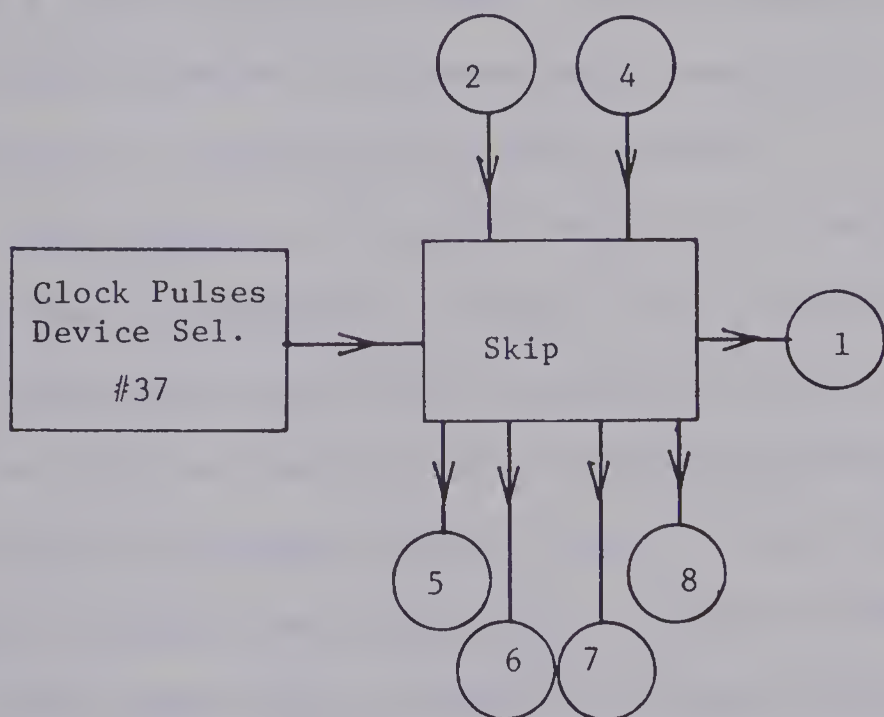


FIG. 3.3 Clock Pulse Control

The main program stops in a check loop at the begining of SKIP which allows the program to continue only on the next clock pulse. The clock pulse is derived from an external square wave generator and monitored by the program by means of device selector 37, using binary code 6371 to load the pulse into the accumulator. Each time the program skips out of this check loop a control variable (ET) of the particular terminal is incremented to provide a record of the elapsed time since the last operation was performed. The pulse generator is set at a rate of 1000 pulses per second to allow sufficient time to complete a control cycle and return to the check loop before a new clock pulse arrives. It was found that the control cycle time never exceeded 800 microseconds, therefore the pulse repetition rate of 1 millisecond is quite adequate.

The remainder of the SKIP subroutine is related to determining what program path to follow in the "central control program". From terminal input data contained in the set of control variables the scan rate selection is checked and recorded, if the terminal is off the program returns to BEGIN. If the terminal is in the process of typing a word the program advances to TWORD. If the terminals switch input has not changed the program advances to the SCAN subroutine. If it has changed the program checks the mode setting and advances to either subroutine MSM or MDM, whichever is applicable. These decisions establish a total of 5 possible program output paths from SKIP and will be discussed at length in Section 3.3.

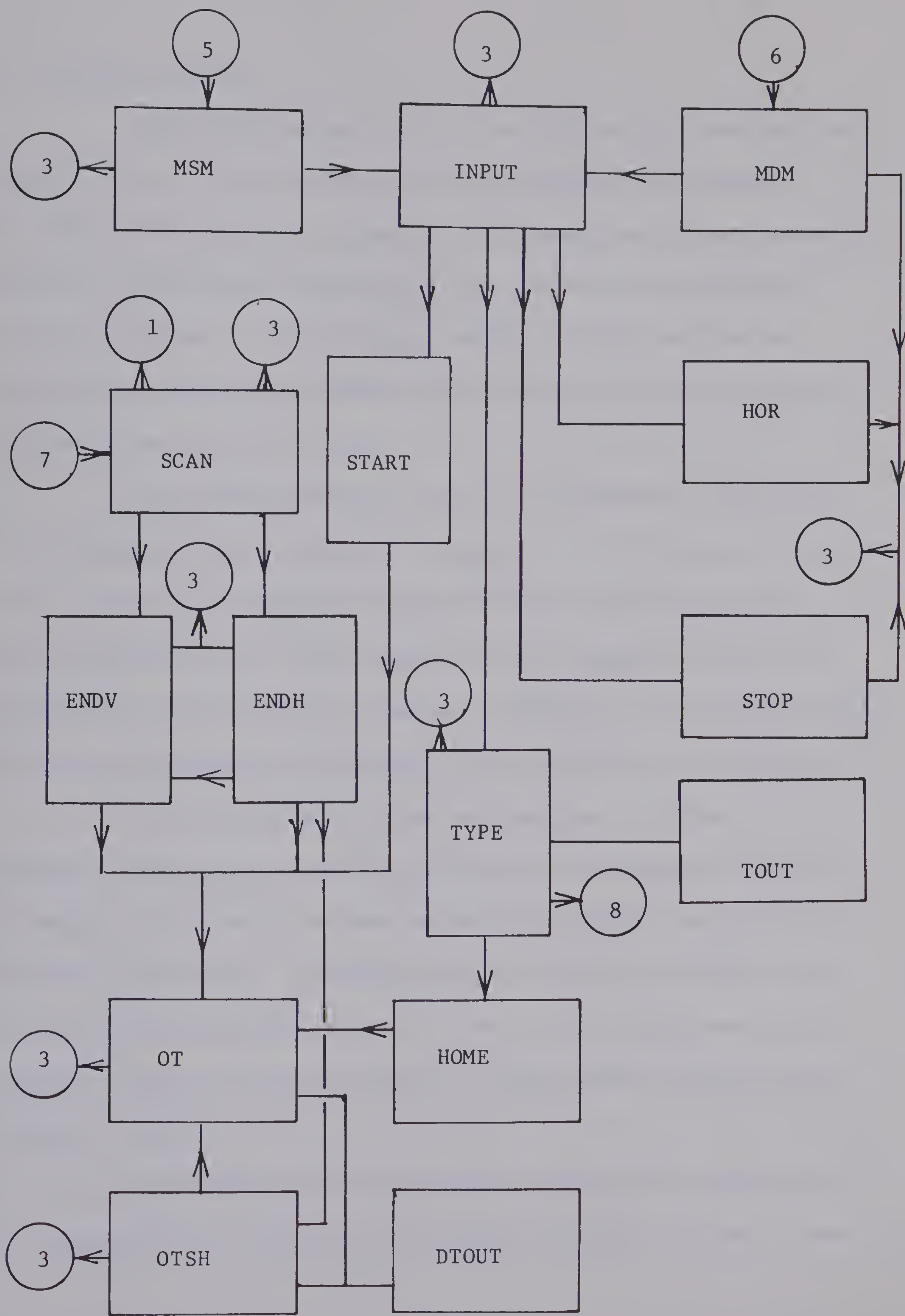


FIG. 3.4 Central Control

3.3 Central Control

The flow diagram for the "central control program" is shown in Fig. 3.4 and consists of 15 subroutines the programs for which can be found in Appendix 'A'. There are a large number of possible program paths depending on the state of the terminals' control variables. The inputs and outputs of this program are designated by encircled numbers which refer to those of other flow diagrams shown in this chapter.

The program inputs 5 and 6 are followed if there has been a change in the terminals input data. If no change has taken place, input 7 is followed to advance the scan location on the basis of elapsed time. The outputs 1 and 3 signify the end of a control cycle and return the program to BEGIN and CLOCK respectively. The output 8 continues the program flow to TWORD shown in Fig. 3.5.

If the skip subroutine has detected a change in the terminals input switch state it will direct the program to MSM if the single acting mode has been selected and MDM if the double acting mode is selected. The MSM subroutine determines whether the switch state has changed from off to on, if so it advances the program to INPUT, if not it ends the control cycle returning the program to CLOCK.

The MDM subroutine determines whether the switch state has changed,if so it advances the program to INPUT, if not it ends the

control cycle returning the program to CLOCK.

The subroutine INPUT is started as the result of an input command from the patients terminal. The program resets ET (elapsed time register) and commences to select the next step in the control sequence. When the display unit is at "home" the program advances to START. Should it be scanning in the vertical direction the program advances to HOR, if scanning in the horizontal direction the program goes to STOP and if stopped the program advances to TYPE. When none of these conditions are satisfied the control cycle is ended and the program returns to BEGIN.

The scan control sequence consists of scanning vertically to the desired row, scanning horizontally to the desired column, stopping at the character and typing it. Each patient input advances the sequence to the next step thereby selecting the character to be typed in an X-Y fashion with four input commands. In the case of the continuous scanning mode the transition from vertical to horizontal scanning is incorporated in the SCAN program automatically and will be discussed later.

The subroutine START interprets the input command as a request to commence vertical scanning. To do so the control variable VS is set to establish a "flag" as to the terminal's position in the control sequence. The scan location (SL) is incremented to "start" and the program is advanced to OT to output the new location to the

terminal display unit. The subsequent program to the end of the control cycle is identical to program paths advancing the scan location and will be described in later paragraphs.

Subroutines HOR and STOP set HS (horizontal scan flag) and SS (stop scan flag) respectively as a result of an input command. HS and SS are terminal control variables which establish the control parameters for the next control cycle. Both subroutines continue on to reset ET and end the control cycle by returning the program to CLOCK.

Should the patient input command request a character to be typed the program advances to subroutine TYPE. This subroutine first checks the scan location to determine if control is to be performed on a character, word or "shift". If "shift" is being displayed control variable SSH is set establishing a control flag to enable upper case typing of the next character scanned. The program then proceeds to HOME which resets SL and advances the program to OT where the display is reset to "home". Should a word be requested the control variable TW is set and the program proceeds to TWORD. If a character is requested the program is advanced through subroutine TOUT which types the character and returns to reset control variables SS, ET and TS. This completes the control cycle and the program returns to CLOCK.

TOUT performs the necessary output transfer to type a character by transmitting IOP4 to the terminal in control. This is accomplished by the binary code 6354 for terminal A and 6364

for terminal B. The terminal in control is detected by monitoring control variables TA and TB for terminals A and B respectively.

If the transition from clock pulse control to central control is on the basis of no change in the terminals input data, the program will be established via the SCAN subroutine of the "central control program". The SCAN subroutine establishes all subsequent program routes to advance the scanning process based on the state of the terminal control variables. The present state of the scan location register (SL) is checked first, if it is at "home" the program is returned to CLOCK thereby ending the control cycle. If the program cycle is to proceed ET is incremented and checked against the scan rate selected, if the time interval has elapsed the program advances to ENDV or ENDH based on whether control variables VS or HS are set. If the time interval has not elapsed the program is returned to CLOCK. The subroutine also monitors the length of time the scan has been stopped at a character. If stopped for three scan rate time intervals the program is advanced to BEGIN resetting the terminal to "home". Each control cycle is recorded by incrementing TS (stop scan elapsed time register) and returning the program to CLOCK until the third cycle is reached.

ENDV is a subroutine that performs three function in the vertical scanning process. To advance SL to the next row, check if the last row has been past and set the parameters for continuous scanning. When the last row has been passed the terminal is reset by returning the program to BEGIN and ending the control cycle. Should a continuous

scanning mode be selected the scanning process is transferred from vertical to horizontal and the program advances to OT. If the scan is to proceed to the next row SL is incremented and the program advances to OT.

Subroutine ENDH performs a similar function in the horizontal mode as ENDV does in the vertical mode. The program starts by incrementing SL to the next column, if the last column has been past the terminal is reset via BEGIN, ending the control cycle. An exception is when the scan is in a continuous mode in which case the terminal is not reset but advanced by directing SL to the first column of the next row via ENDV. When SL is at the "start" position the program advances to HOME to reset SL and reset the display unit to "home" by means of OT. Should the new scan location be "shift" the program proceeds to OTSH to illuminate the display lamp "shift". If the new location is any other character the program advances to OT to output the new location to the display unit.

Subroutine OT, in conjunction with DOUT, performs the necessary output transfer of SL to the terminal to display the new location. In addition OT also masks out bits 0 and 1 of SL, conditional on the state of control variable SSH, to eliminate illuminating shift other than when SSH is set. The subroutine ends by resetting ET and returning the program to CLOCK ending the control cycle.

Subroutine OTSH performs a similar function to OT but only in the case of scan being advanced to "shift". The program starts by checking whether SSH is set, if so the program is transferred to OT indicating that the "shift" lamp is already set. If SSH is not set the program proceeds to set SL with SHL (shift lamp location) and outputs the new location to the terminal by DOUT. The program returns from DOUT resets ET and proceeds to end the control cycle by returning to CLOCK.

DOUT performs the necessary output transfer to illuminate a display location by transmitting IOP2 to the terminal in control. This is accomplished by the binary code 6352 for terminal A and 6362 for terminal B. The terminal in control is detected by monitoring control variables TA and TB for terminals A and B respectively.

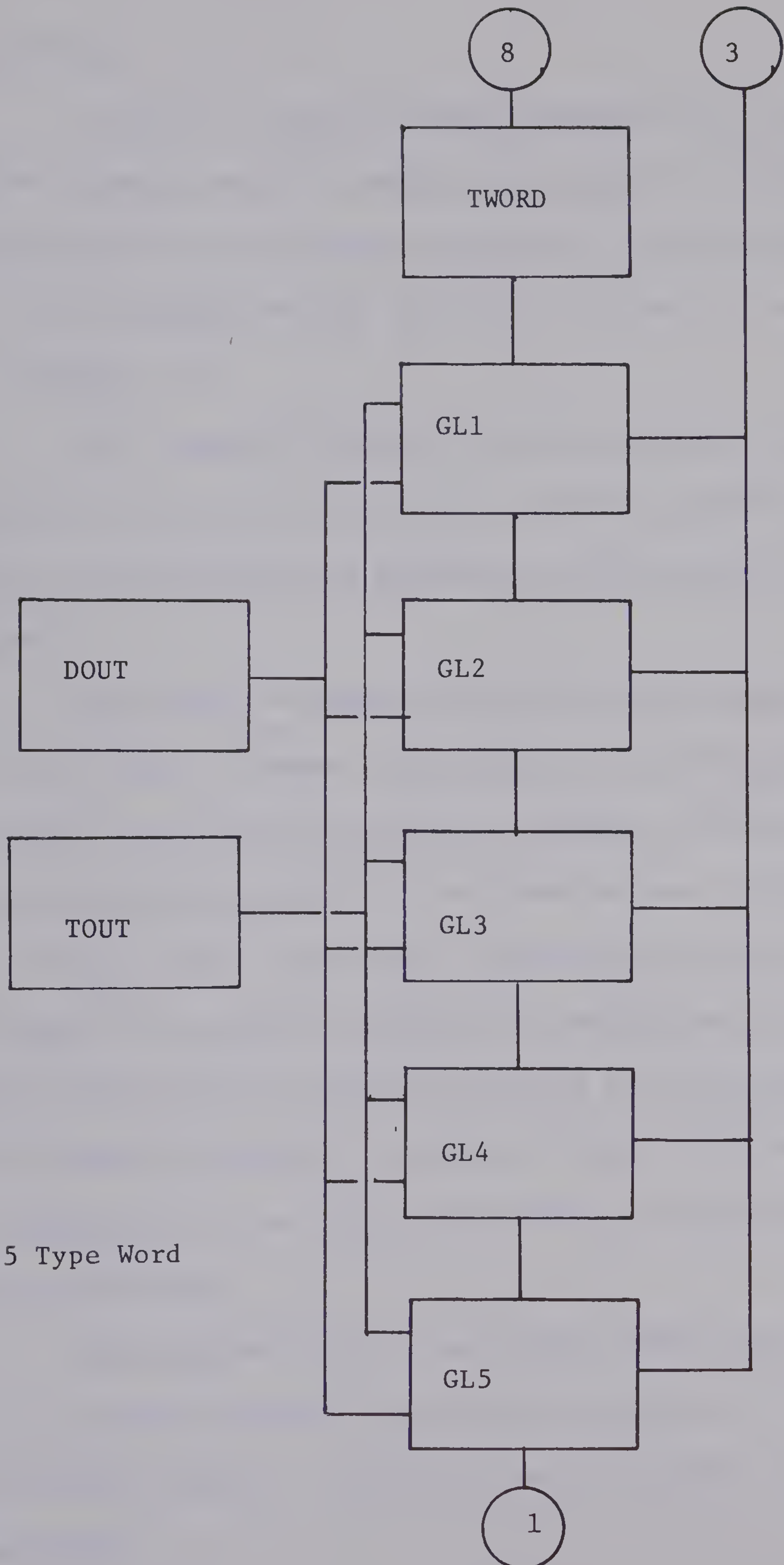


FIG. 3.5 Type Word

3.4 Type Word

In order to type the words depicted on the last row of the terminal display unit the program "type word" is provided. This program consists of 8 subroutines shown in the flow diagram of Fig. 3.5. The programs for the individual subroutines can be found in Appendix 'A'.

The "type word" control program requires 6 control cycles to completely type the word requested. The first control cycle sets the letter of the word and the subsequent cycles type the letters plus a space.

The control variable TW is set by the "central control program" as a "flag" to divert the program to the "type word" program on each control cycle until the word is complete. WS is a control variable, set after the letters to be typed are set, to allow the program to skip to the "type letter" subroutines for the remaining control cycles. The control variable RTW establishes the rate at which the letters will by typed by comparing it to the elapsed time register ET before proceeding to type the letter. The word typing rate is independent of the "scan rate" setting, being set at 5 characters per second.

The program starts at subroutine TWORD which sets the letters of the word requested, sets WS and proceeds to GL1 subroutine to type the first letter. With WS set each subsequent control cycle by-passes TWORD.

Subroutine GL1 increments ET. If the typing time interval has elapsed the program proceeds to type the first letter, if not the program returns to CLOCK ending the control cycle. Should the first letter already be typed the program advances to GL2 to type the second letter.

The letters typed by subroutines GL1 through GL5 are stored as control variables L1 through L5, set by TWORD. Each letter is typed by its respective subroutine in an identical manner. The scan location control variable SL is set with the letter and displayed on the terminal's display by means of DOUT (previously discussed). The letter variable and ET are reset to prepare for the next control cycle. The letter is finally typed by means of subroutine TOUT (previously discussed) and the program is returned to CLOCK thereby ending the control cycle.

Subroutine GL5, in addition to typing letter 5 as above, resets control variables TW and WS and returns the program to subroutine BEGIN to reset the terminal to the "home" position.

CONCLUSION

The project described successfully controlled two typewriter terminals by means of a computer, and in so doing demonstrated the feasibility of establishing much larger and more sophisticated systems. Future work in this field could develop more elaborate programs which would increase patient typing speed and produce options enabling him to perform other useful functions.

The process of scanning for individual letters is relatively slow. The seven words which were included show the advantage of a system capable of typing complete words. A large library could be programmed into the computer, with access by means of numerical codes. The patient would use a display consisting of numbers, and by selecting appropriate number sequences he could type complete words.

Although the present system provides a means by which selection errors can be avoided many errors still occur. This is primarily due to the fact that the typewritten page cannot be readily observed. A CRT (cathode ray tube) could be connected to the computer in order that a complete sentence or text could be formed and corrected before typing. For economy the CRT could display a sentence for each terminal simultaneously in full view of the patients.

Another means by which typing speed could be increased would be to employ Morse Code techniques. If the patient has sufficient control to initiate codes pulses the computer could detect them and

type the character without the aid of a display unit.

Peripheral devices can be controlled by this system as exhibited by the SW (switch) location on the display unit. A display could be developed to control a large number of external mechanical devices, enabling the patient to perform various useful functions.

REFERENCES

1. Norbert L. Schmitz & Donald W. Novotny — Introductory Electromechanics, The Ronald Press Co., 1965.
2. Samuel Seely, Ph.D., — Electromechanical Energy Conversion, McGraw-Hill Book Co. Inc., 1962.
3. Digital Equipment Corporation, — Introduction to Programming, Magnard, Massachussetts, U.S.A., 1968.
4. David Kohn, — Codebreakers, p. 100-101, The Macmillan Co., 1967.
5. Helen Fouche Gaines, — Cryptanalysis, p. 218-227, Dover Publications Inc., 1956.
6. Richard Kildaw, — A Multi-terminal Interface Allowing Typewriter Operation by Paraplegics (MITOP), Department of Electrical Engineering, University of Alberta, M.Sc. Thesis, 1969.
7. R.K.C. Paau, — A Typewriting System For The Seriously Handicapped, Department of Electrical Engineering, University of Toronto, M.Sc. Thesis, 1969.
8. A.F. Newell & C.D. Nabaui, — VOTEM: The Voice Operated Typewriter Employing Morse Code, Journal of Scientific Instruments, (Journal of Physics E.) Ser. 2, Vol. 2., 1969.
9. Digital Equipment Corporation, — Logic Handbook, Magnard, Massachussetts, U.S.A., Positive Logic Edition, 1969.
10. William E. Wickes, — Logic Design With Integrated Circuits, John Wiley and Sons Inc., 1968.

APPENDIX 'A'

SYSTEMS PROGRAM CTSH

```
*20                 /CONTROL VARIABLES

CONTROL,0020        /START ADD. LOAD-STORE CONTROL VARIABLES
SSH,     0000       /SET SHIFT
TS,      0000       /TIME STOP
SI,      0000       /STATE INPUT
NCSS,    0000       /NEW CONTROL SWITCH STATE
SL,      0000       /SCAN LOCATION
ET,      0000       /ELAPSED TIME
VS,      0000       /VERTICAL SCAN
HS,      0000       /HORIZONTAL SCAN
SS,      0000       /STOP SCAN
OI,      0000       /OLD INPUT
NI,      0000       /NEW INPUT
SM,      0000       /SINGLE ACTING MODE
DM,      0000       /DOUBLE ACTING MODE
CSM,     0000       /CONTINUOUS SM
CDM,     0000       /CONTINUOUS DM
TW,      0000       /TYPE WORD
WS,      0000       /WORD SET
SR,      0000       /SCAN RATE
L1,      0000       /LETTER 1
L2,      0000       /LETTER 2
L3,      0000       /LETTER 3
L4,      0000       /LETTER 4
L5,      0000       /LETTER 5
M1,      0370       /MASK 1
M2,      0400       /MASK 2
M3,      1770       /MASK 3
M4,      0070       /MASK 4
M5,      1700       /MASK 5
M6,      0017       /MASK 6
M7,      0030       /MASK 7
M8,      0010       /MASK 8
M9,      0020       /MASK 9
FR,      0020       /FIRST ROW
WR,      0070       /WORD ROW
SHLS,    4000       /SHIFT LAMP SET
SHL,     0120       /SHIFT LOCATION
VI,      0010       /VERTICAL INCREMENT
EV,      0100       /END VERTICAL SCAN
HI,      0100       /HORIZONTAL INCREMENT
EH,      1300       /END HORIZONTAL SCAN
MSR,     0340       /MASK SCAN RATE
HST,     0110       /HORIZONTAL ON START
TA,      0000       /TERMINAL A
TB,      0000       /TERMINAL B
NVAR,    -27        /NUMBER OF VARIABLES
COUNT,   -27        /VARIABLE COUNT
WSS,     1220       /WORD SET SUBROUTINE
RTW,     0050       /RATE TYPE WORD
LT,      0330       /LETTER T
LH,      0160       /LETTER H
LE,      0420       /LETTER E
LO,      0150       /LETTER O
```

```
LF,     0750     /LETTER F
LA,     0240     /LETTER A
LN,     0520     /LETTER N
LD,     0530     /LETTER D
LI,     0430     /LETTER I
LS,     0340     /LETTER S
LR,     0250     /LETTER R
SP,     0130     /SPACE

*200             /PROGRAM CCTSH STARTING ADDRESS

HLT              /HALT BEFORE PROCEEDING

BEGIN,  CLA      /RESET ALL VARIABLES
TAD     TA
SZA CLA
JMP     ABEGIN
TAD     TB
SZA CLA
JMP     BBEGIN
JMP     NCLOCK

ABEGIN, CLA      /RESET TERMINAL A
TAD     CONTROL
DCA     10
DCA I   10
ISZ     COUNT
JMP     ABEGIN+3
CLA
TAD     NVAR
DCA     COUNT
6307
6352
6313
6352
DCA     01
JMP     SETM

BBEGIN, CLA      /RESET TERMINAL B
TAD     CONTROL
DCA     10
DCA I   10
ISZ     COUNT
JMP     BBEGIN+3
CLA
TAD     NVAR
DCA     COUNT
6307
6362
6361
6313
6362
DCA     01
JMP     SETM
```

```
SETM,   CLA      /SET SCAN MODE OF TERMINAL IN CONTROL
TAD     01
AND     M7
SNA CLA
JMP     .+14
TAD     01
AND     M8
SNA CLA
JMP     .+13
TAD     01
AND     M9
SNA CLA
JMP     .+12
IAC
DCA     DM
JMP     NCLOCK
IAC
DCA     SM
JMP     NCLOCK
IAC
DCA     CSM
JMP     NCLOCK
IAC
DCA     CDM
JMP     NCLOCK

ALOAD,  0000     /LOAD CONTROL VARIABLES TERMINAL A
CLA
TAD     ATERM
DCA     10
TAD     CONTROL
DCA     11
TAD I   10
DCA I   11
ISZ     COUNT
JMP     ALOAD+6
CLA
TAD     NVAR
DCA     COUNT
JMP I   ALOAD

BLOAD,  0000     /LOAD CONTROL VARIABLES TERMINAL B
CLA
TAD     BTERM
DCA     10
TAD     CONTROL
DCA     11
TAD I   10
DCA I   11
ISZ     COUNT
JMP     BLOAD+6
CLA
TAD     NVAR
DCA     COUNT
JMP I   BLOAD
```

```
ASTORE, 0000    /STORE CONTROL VARIABLES TERMINAL A
CLA
TAD     ATERM
DCA     10
TAD     CONTROL
DCA     11
TAD I   11
DCA I   10
ISZ     COUNT
JMP     ASTORE+6
CLA
TAD     NVAR
DCA     COUNT
JMP I   ASTORE

BSTORE, 0000    /STORE CONTROL VARIABLES TERMINAL B
CLA
TAD     BTERM
DCA     10
TAD     CONTROL
DCA     11
TAD I   11
DCA I   10
ISZ     COUNT
JMP     BSTORE+6
CLA
TAD     NVAR
DCA     COUNT
JMP I   BSTORE

*400            /START PAGE 2

ATERM,  400     /CONTROL VARIABLES TERMINAL A
ASSH,   0000    /SET SHIFT TERMINAL A
ATS,    0000    /TIME STOP TERMINAL A
ASI,    0000    /STATE INPUT TERMINAL A
ANCSS,  0000    /NEW CONTROL SWITCH STATE TERMINAL A
ASL,    0000    /SCAN LOCATION TERMINAL A
AET,    0000    /ELAPSED TIME TERMINAL A
AVS,    0000    /VERTICAL SCAN TERMINAL A
AHS,    0000    /HORIZONTAL SCAN TERMINAL A
ASS,    0000    /STOP SCAN TERMINAL A
AOI,    0000    /OLD INPUT TERMINAL A
ANI,    0000    /NEW INPUT TERMINAL A
ASM,    0000    /SINGLE ACTING MODE TERMINAL A
ADM,    0000    /DOUBLE ACTING MODE TERMINAL A
ACSM,   0000    /CONTINUOUS SM TERMINAL A
ACDM,   0000    /CONTINUOUS DM TERMINAL A
ATW,    0000    /TYPE WORD TERMINAL A
AWS,    0000    /WORD SET TERMINAL A
ASR,    0000    /SCAN RATE TERMINAL A
AL1,    0000    /LETTER 1 TERMINAL A
AL2,    0000    /LETTER 2 TERMINAL A
AL3,    0000    /LETTER 3 TERMINAL A
AL4,    0000    /LETTER 4 TERMINAL A
```

```
BTERM,  0430    /CONTROL VARIABLES TERMINAL B
BSSH,   0000    /SET SHIFT TERMINAL B
BTS,    0000    /TIME STOP TERMINAL B
BSI,    0000    /STATE INPUT TERMINAL B
BNCSS,  0000    /NEW CONTROL SWITCH STATE TERMINAL B
BSL,    0000    /SCAN LOCATION TERMINAL B
BET,    0000    /ELAPSED TIME TERMINAL B
BVS,    0000    /VERTICAL SCAN TERMINAL B
BHS,    0000    /HORIZONTAL SCAN TERMINAL B
BSS,    0000    /STOP SCAN TERMINAL B
BOI,    0000    /OLD INPUT TERMINAL B
BNI,    0000    /NEW INPUT TERMINAL B
BSM,    0000    /SINGLE ACTING MODE TERMINAL B
BDM,    0000    /DOUBLE ACTING MODE TERMINAL B
BCSM,   0000    /CONTINUOUS SM TERMINAL B
BCDM,   0000    /CONTINUOUS DM TERMINAL B
BTW,    0000    /TYPE WORD TERMINAL B
BWS,    0000    /WORD SET TERMINAL B
BSR,    0000    /SCAN RATE TERMINAL B
BL1,    0000    /LETTER 1 TERMINAL B
BL2,    0000    /LETTER 2 TERMINAL B
BL3,    0000    /LETTER 3 TERMINAL B
BL4,    0000    /LETTER 4 TERMINAL B
BL5,    0000    /LETTER 5 TERMINAL B

NCLOCK, CLA     /START CLOCK FROM BEGIN
TAD     TA
SZA CLA
JMP     .+20
TAD     TB
SZA CLA
JMP     .+2
JMP     .+10
CLA
DCA     TB
IAC
DCA     TA
JMS     BSTORE
JMS     ALOAD
JMP     .+13
CLA
IAC
DCA     TA
JMP     ABEGIN
CLA
DCA     TA
IAC
DCA     TB
JMS     ASTORE
JMS     BLOAD
CLA
TAD     OI
AND     MRS
DCA     SR
JMP     SKIP
```

```
CLOCK,  CLA       /SET VARIABLES FOR NEXT CLOCK SKIP
TAD     N1
DCA     O1
TAD     TA
SZA CLA
JMP     .+7
DCA     TB
IAC
DCA     TA
JMS     BSTORE
JMS     ALOAD
JMP     .+7
CLA
DCA     TA
IAC
DCA     TB
JMS     ASTORE
JMS     BLOAD
CLA

*600              /START PAGE 3

SKIP,   6371      /WAIT SKIP OUT ON CLOCK PULSE
JMP     .-1
6372
CLA
TAD     O1
AND     MSR
DCA     SR
TAD     SR
SNA CLA
JMP     BEGIN
TAD     TW
SZA CLA
JMP     TWORD
JMS     MODE
CLA
TAD     N1
CIA
TAD     O1
SNA CLA
JMP     SCAN
TAD     N1
AND     M1
DCA     NCSS
TAD     O1
AND     M1
CIA
TAD     NCSS
SZA CLA
JMP     BEGIN
TAD     SM
SZA CLA
JMP     MSM
TAD     DM
```

```
                        /SKIP CONTINUED

SZA CLA
JMP         MDM
TAD         CSM
SZA CLA
JMP         MSM
TAD         CDM
SZA CLA
JMP         MDM
JMP         BEGIN

MODE,       0000        /INPUT CONTROL MODE
CLA
TAD         SL
6307
TAD         1A
SZA CLA
JMP         .+6
6361
6313
6362
DCA         N1
JMP         .+6
CLA
6351
6313
6352
DCA         N1
JMP I       MODE

MSM,        CLA         /SINGLE ACTING MODE SCAN 0-1
TAD         N1
AND         M2
DCA         S1
TAD         01
AND         M2
CIA
TAD         S1
SPA SNA
JMP         CLOCK
JMP         INPUT

MDM,        CLA         /DOUBLE ACTING MODE SCAN 0-1,1-0
TAD         N1
AND         M2
DCA         S1
TAD         01
AND         M2
CIA
TAD         S1
SZA CLA
JMP         INPUT
JMP         CLOCK
```

```
INPUT,   CLA      /NEW INPUT CONTROL SCAN
DCA      ET
TAD      SL
AND      M3
SNA CLA
JMP      START
TAD      VS
SZA CLA
JMP      HOR
TAD      HS
SZA CLA
JMP      STOP
TAD      SS
SNA CLA
JMP      BEGIN
JMP      TYPE

*1000             /START PAGE 4

SCAN,    CLA      /ADVANCE TO NEXT LETTER
TAD      SL
AND      M3
SNA CLA
JMP      CLOCK
TAD      ET
IAC
DCA      ET
TAD      ET
CIA
TAD      SR
SZA CLA
JMP      CLOCK
TAD      VS
SZA CLA
JMP      ENDV
TAD      HS
SZA CLA
JMP      ENDH
TAD      TS
IAC
DCA      TS
TAD      TS
CIA
TAD      TSS
SNA CLA
JMP      BEGIN
TAD      SS
SNA CLA
JMP      BEGIN
DCA      ET
JMP      CLOCK
```

```
TYPE,   CLA       /TYPE REQUEST, CHECK AND RESET
TAD     SHL
CIA
TAD     SL
AND     M3
SZA CLA
JMP     .+5
IAC
DCA     SSH
DCA     SS
JMP     HOME
CLA
TAD     SL
AND     M4
CIA
TAD     WR
SZA CLA
JMP     .+4
IAC
DCA     TW
JMP     TWORD
JMP     TOUT
CLA
DCA     SS
DAC     ET
DCA     TS
JMP     CLOCK

TOUT,   0000      /IOT TYPE CHARACTER
CLA
TAD     TA
SZA CLA
JMP     .+3
6364
JMP     .+2
6354
JMP I   TOUT

ENDV,   CLA       /VERTICAL SCAN REQUEST CHECK
TAD     SL
TAD     V1
DCA     SL
TAD     SL
AND     M3
CIA
TAD     EV
SNA CLA
JMP     BEGIN
TAD     CSM
TAD     CDM
SNA CLA
JMP     OT
TAD     SL
AND     M3
CIA
```

```
TAD     FR
SZA CLA
JMP     OT
DCA     VS
IAC
DCA     HS
JMP     OT

ENDH,   CLA          /HORIZONTAL SCAN REQUEST CHECK
TAD     SL
TAD     HI
DCA     SL
TAD     SL
AND     M3
CIA
TAD     EH
SPA CLA
JMP     .+16
TAD     SL
AND     M3
CIA
TAD     HST
SNA CLA
JMP     HOME
TAD     SL
AND     M3
CIA
TAD     SHL
SNA CLA
JMP     OTSH
JMP     OT
TAD     CSM
TAD     CDM
SNA CLA
JMP     BEGIN
TAD     SL
AND     M4
DCA     SL
JMP     ENDV

*1200                /START PAGE 5

OT,     CLA          /ADVANCE SCAN REQUEST
TAD     SSH
SZA CLA
JMP     .+4
TAD     SL
AND     M3
DCA     SL
JMS     DOUT
CLA
DCA     ET
JMP     CLOCK
```

```
OTSH,   CLA     /DISPLAY SHIFT REQUEST
TAD     SSH
SZA CLA
JMP     OT
TAD     SL
TAD     SHL
DCA     SL
JMS     DOUT
CLA
DCA     ET
JMP     CLOCK

DOUT,   0000    /IOT NEW DISPLAY LOCATION
CLA
TAD     TA
SZA CLA
JMP     .+5
TAD     SL
6307
6362
JMP     .+5
CLA
TAD     SL
6307
6352
JMP I   DOUT

HOME,   CLA     /RESET TO HOME REQUEST
DCA     SL
TAD     SSH
SNA CLA
JMP     OT
TAD     SHLS
DCA     SL
JMP     OT

START,  CLA     /ADVANCE SCAN TO START AND SET VS
IAC
DCA     VS
TAD     SL
TAD     VI
DCA     SL
JMP     OT

HOR,    CLA     /SET HS
IAC
DCA     HS
DCA     VS
DCA     ET
JMP     CLOCK
```

```
STOP,   CLA      /SET SS
IAC
DCA     SS
DCA     HS
DCA     ET
JMP     CLOCK

TWORD,  CLA      /TYPE WORD REQUEST
TAD     WS
SZA CLA
JMP     GL1
TAD     SL
AND     M5
RAR
RAR
RAR
RAR
RAR
RAR
AND     M8
TAD     WSS
DCA     AD1
JMP I   AD1
JMP     GL5
JMP     THE
JMP     OF
JMP     WAND
JMP     TO
JMP     IN
JMP     THAT
JMP     IS
JMP     IT
JMP     FOR
JMP     AS
AD1,    0000     /ADDRESS STORE

THE,    CLA      /SET LETTERS FOR 'THE'
TAD     LT
DCA     L1
TAD     LH
DCA     L2
TAD     LE
DCA     L3
TAD     SP
DCA     L4
IAC
DCA     WS
JMP     GL1
```

```
OF,     CLA     /SET LETTERS FOR 'OF'
TAD     LO
DCA     L1
TAD     LF
DCA     L2
TAD     SP
DCA     L3
IAC
DCA     WS
JMP     GL1

*1400           /START PAGE 6

WAND,   CLA     /SET LETTERS FOR 'AND'
TAD     LA
DCA     L1
TAD     LN
DCA     L2
TAD     LD
DCA     L3
TAD     SP
DCA     L4
IAC
DCA     WS
JMP     GL1

TO,     CLA     /SET LETTERS FOR 'TO'
TAD     LT
DCA     L1
TAD     LO
DCA     L2
TAD     SP
DCA     L3
IAC
DCA     WS
JMP     GL1

IN,     CLA     /SET LETTERS FOR 'IN'
TAD     LI
DCA     L1
TAD     LN
DCA     L2
TAD     SP
DCA     L3
IAC
DCA     WS
JMP     GL1

THAT,   CLA     /SET LETTERS FOR 'THAT'
TAD     LT
DCA     L1
TAD     LH
DCA     L2
TAD     LA
DCA     L3
```

```
                        /THAT CONTINUED

TAD       LT
DCA       L4
TAD       SP
DCA       L5
IAC
DCA       WS
JMP       GL1

IS,       CLA           /SET LETTERS FOR 'IS'
TAD       LI
DCA       L1
TAD       LS
DCA       L2
TAD       SP
DCA       L3
IAC
DCA       WS
JMP       GL1

IT,       CLA           /SET LETTERS FOR 'IT'
TAD       LI
DCA       L1
TAD       LS
DCA       L2
TAD       SP
DCA       L3
IAC
DCA       WS
JMP       GL1

FOR,      CLA           /SET LETTERS FOR 'FOR'
TAD       LF
DCA       L1
TAD       LO
DCA       L2
TAD       LR
DCA       L3
TAD       SP
DCA       L4
IAC
DCA       WS
JMP       GL1

AS,       CLA           /SET LETTERS FOR 'AS'
TAD       LA
DCA       L1
TAD       LS
DCA       L2
TAD       SP
DCA       L3
IAC
DCA       WS
JMP       GL1
```

```
*1600           /START PAGE 7

GL1,    CLA     /TYPE FIRST LETTER
TAD     E1
IAC
DCA     E1
TAD     E1
CIA
TAD     RN1
SZA CLA
JMP     CLOCK
TAD     L1
SNA CLA
JMP     GL2
TAD     SSH
SNA CLA
JMP     .+2
TAD     SHLS
TAD     L1
DCA     SL
JMS     DTOUT
CLA
DCA     L1
DCA     E1
JMS     TOUT
JMP     CLOCK

GL2,    CLA     /TYPE SECOND LETTER
TAD     L2
SNA CLA
JMP     GL3
TAD     L2
DCA     SL
JMS     DOUT
CLA
DCA     L2
DCA     E1
CLA
JMS     TOUT
JMP     CLOCK

GL3,    CLA     /TYPE THIRD LETTER
TAD     L3
SNA CLA
JMP     GL4
TAD     L3
DCA     SL
JMS     DOUT
CLA
DCA     L3
DCA     E1
JMS     TOUT
JMP     CLOCK
```

```
GL4,    CLA      /TYPE FORTH LETTER
TAD     L4
SNA CLA
JMP     GL5
TAD     L4
DCA     SL
JMS     DOUT
CLA
DCA     L4
DCA     E1
JMS     TOUT
JMP     CLOCK

GL5,    CLA      /TYPE FIFTH LETTER
TAD     L5
SNA CLA
JMP     .+11
TAD     L5
DCA     SL
JMS     DOUT
CLA
DCA     L5
DCA     E1
JMS     TOUT
JMP     CLOCK
CLA
DCA     E1
DCA     TW
DCA     WS
JMP     BEGIN
```